Development of a Control Scheme for a Quarter Car Test Rig

by

© Mehedi Ehteshum

A thesis submitted to the School of Graduate Studies in partial fulfillment of the requirements for the degree of

Master of Engineering

Faculty of Engineering and Applied Science

Memorial University of Newfoundland

May 2018

St. John's, Newfoundland and Labrador

Abstract

Road holding performance and vibration isolation of an automobile are some of the most important criteria for human perception of ride quality. For this, the accurate estimation of car body vibration is a prerequisite. A 2-DOF quarter car is a simple, but still reasonable approach to study the dynamic behaviour of a car body. This thesis illustrates the development of control scheme hardware for a quarter car model based on an existing test rig. The test rig parameters are estimated using different static and dynamic tests. The parameters are then used to develop a passive nonlinear model for the test rig. A similar linear model is used to develop an idealized controller. The controller is then applied to the nonlinear simulation model to make it active, and its performance is found to be slightly better at low frequencies for the nonlinear model. The actuator dynamics are then included in the active model to make it realistic. A comparative study of the ideal and realistic model shows that the realistic active model generally shows better ride quality, specially at high frequencies. This model will offer the future researchers a realistic control scheme for the quarter car test rig. The thesis shows the implementation of a new software (20-sim 4C) and hardware system to allow control signals to be sent from the simulation software to the physical quarter car, and verification of the software

and hardware by controlling a voice coil actuator using pulse-width modulation to follow a position command signal and then a force command signal. It is observed that as the frequency of the command signal increases, the amplitude loss in the response increases. The actuator can generate only about 5 N force when the frequency of the force command signal is above about 10 Hz. A more robust actuator with higher bandwidth will be required for hardware replication of the maximum potential active suspension benefit predicted by the simulation models.

Acknowledgements

Foremost, I would like to thank the Almighty.

Then I would like to express my heartiest gratitude and respect to a great human being, Dr. Geoff Rideout, with whom I have walked the path of my life for a while and learned along the way. Surely, I am not the best of his students, but I can't imagine a better supervisor and teacher than him. I tried to follow him and now I regret that I could do better use of my potentials with the help of him, but time is irreversible.

I would like to gratefully thank my loved ones, who have supported me throughout the entire process by keeping me harmonious.

Last but not the least, I would like to sincerely thank all the people and lab technologists who have helped me throughout my work, especially Dr. Oscar De Silva, Tom Pike, Brian Pretty, Steve Steele, Greg O'Leary, Matt Curtis, Jason Murphy, Patrick, Jerry, Tu Tran, and late Glenn St. Croix who never failed to refresh me with his smile every time I went past him.

Table of Contents

List of Tables

List of Figures

Nomenclature

ϕ_i	Mode shapes of the system
ω_i	Natural frequencies of the system
A	Linear 2-DOF quarter car model system matrix
B	Linear 2-DOF quarter car model disturbance matrix
b_1	Rail friction coefficient for sprung mass motion (N.s/m)
b_2	Rail friction coefficient for unsprung mass motion (N.s/m)
b_s	Suspension damping coefficient (N.s/m)
C	Viscous damping coefficient matrix
D	Linear 2-DOF quarter car model gravity matrix
F_a	Actuator force (N)
F_d	Suspension damping force (N)
F_{f1}	Rail friction for unsprung mass motion (N)
F_{f2}	Rail friction for sprung mass motion (N)
G	Feedback gain
G_1	Gain of AD620 instrumentation amplifier
g	Gravitational acceleration (m/s^2)

H_A	Acceleration transfer function
H_{RS}	Rattle space transfer function
H_{TD}	Tire deflection transfer function
I	Identity matrix
J	Performance index
K	Stiffness matrix
k_1	Suspension spring 1 (Red) stiffness (N/m)
k_2	Suspension spring 2 (Black) stiffness (N/m)
k_s	Suspension stiffness (N/m)
k_t	Tire stiffness (N/m)
L	Linear 2-DOF quarter car model road input matrix
M	Linear 2-DOF quarter car model mass matrix
m_s	Sprung mass (kg)
m_u	Unsprung mass (kg)
N	LQR input-state combination weighting matrix
P	Mass-normalized modal matrix
p_i	Performance index weighting term
Q	LQR state combination weighting matrix
R	LQR input combination weighting matrix
R_G	Gain resistor for AD620 instrumentation amplifier
u	Input matrix

v_s	Sprung mass absolute velocity (m/s)
v_{susp}	Suspension absolute velocity (m/s)
v_u	Unsprung mass absolute velocity (m/s)
x	Linear 2-DOF quarter car model state variable matrix
z	Linear 2-DOF quarter car model absolute position matrix
z_s	Sprung mass absolute position (m)
z_{susp}	Suspension absolute deflection (m)
z_u	Unsprung mass absolute position (m)
z_r	Absolute road profile (m)
z_t	Tire absolute deflection (m)

Chapter 1: Introduction

Automobiles travelling at a high speed are subjected to a wide spectrum of vibrations which can be classified as 'ride' (0-25 Hz) and 'noise' (25-20,000 Hz). Excitation sources include road roughness, tire/wheel, driveline and engine. People's perception of comfort depends largely on the isolation of these vibrations [1].

1.1 Suspension Objectives

Vehicle suspension serves for the aforementioned purpose. It supports the chassis on the axles and contributes to the car's handling and ride comfort. The basic tasks of an automotive suspension are as follows [2]-

1. Isolation of a car body from road disturbances to provide good ride quality. Vehicle body acceleration quantifies ride quality.
2. Maintaining good road holding. This performance can be characterized in terms of vehicle's braking, cornering and traction abilities. It can be quantified by the tire

deflection, as a tire can be compared to a high stiffness compressive spring because of its stiffness.

3. Providing good handling. It can be quantified by the roll and pitch accelerations of a vehicle during braking, cornering and traction.
4. Supporting the vehicle static weight. It can be quantified by the rattle space or the suspension deflection.

A designer needs to deal with these conflicting demands in designing the vehicle suspension system [3].

1.2 Application of Quarter Car Test Rig

For the reason mentioned above, the study of vehicle dynamic model is so important in vehicle dynamics. Compared to other possible dynamic vehicle models: full car and half car, a quarter car model is the simplest as well as it gives reasonable results. The study of quarter car model is important because of its analytical and computational simplicity, and its predictive ability despite that.

A 2-DOF quarter car model is a systematic treatment of a vehicle as a dynamic system that can represent the actual motions of the body and axles in response to the road profile. It consists of a sprung mass (m_s) that represents the one-fourth of a car body and unsprung mass (m_u) that is the equivalent mass of the axles and associated wheel and other hardware not supported by the suspension. The suspension consists of a spring (k_s), a damper (b_s) and (in an active suspension) an active force (F_a) actuator. For a passive

suspension, the actuator is not included in the model. A simple spring (k_t) below the unsprung mass represents the tire stiffness which often includes a damper [4, 5]. Figure 1.1 shows the schematic of a quarter car model.

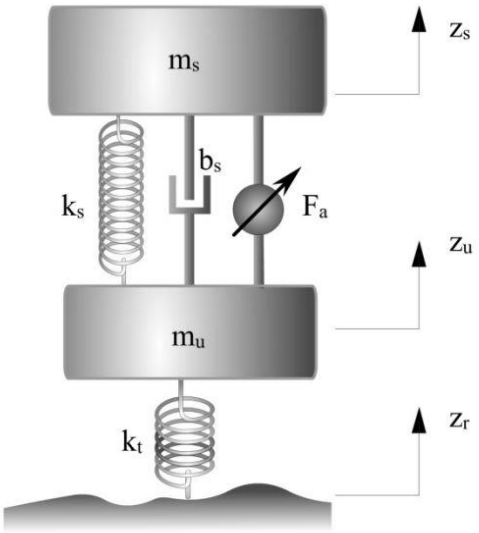


Figure 1.1: Quarter car active suspension model [2].

Performance of a vehicle suspension system is generally assessed in terms of sprung mass acceleration (ride quality), suspension deflection (rattle space) and tire deflection (road holding) [3].

1.3 Influence of Suspension Parameters

A suspension with low stiffness reduces sprung mass acceleration at high frequency road inputs. However, suspension deflection increases at low frequencies in this case. High suspension damping reduces or eliminates the first resonance of the sprung mass and the suspension deflection but increases harshness at high frequencies. High stiffness tire

reduces tire deflection at low frequencies. However, it increases the second resonance [2].

1.4 Active Suspension and Control Benefits

These trade-offs in performance discussed above between the ride quality, rattle space and road holding can be improved by using active or semi-active suspension instead of using conventional passive suspension consisting of spring and shock absorber [2]. The 'active' qualities in suspensions can be classified as follows [1]-

1. Passive suspension. This type has no external energy directly supplied to it.
2. Self-leveling suspension. It can adjust for changes in load to bring the suspension deflection within the desired range. As the pressure changes with load, its stiffness changes as well to keep its natural frequency constant all the time.
3. Semi-active suspension. Its stiffness and damping properties can be changed by an external control.
    - Slow-active suspension. The stiffness and damping properties of this suspension can be switched between several discrete levels depending on the changes in driving conditions. It can change to a higher level of damping and/or stiffness within a fraction of a second, but it delays during switching back to a softer setting.
    - Low-bandwidth suspension. It can adjust itself continuously in response to the low-frequency sprung mass motions (1-3 Hz).

- High-bandwidth suspension. It can adjust itself continuously in response to the low-frequency sprung mass motions (1-3 Hz) as well as the high-frequency axle motions (10-15 Hz).

4. Full-active suspension. It incorporates an actuator that can not only dissipate energy but also supply energy. The actuator can be either hydraulic, pneumatic or electric which requires external power. It also has low-bandwidth and high-bandwidth variants.

The active or semi-active control of suspension allows it to enhance the ride performance by sensing and controlling pitch and bounce motions without compromising the handling behaviour. It can control vehicle height within the design limits despite changes in load or aerodynamic forces. It eliminates the changes in handling that could happen due to the height change. It reduces the ride height for reduced drag at high speeds or alters pitch attitude to modify aerodynamic lift. It can increase damping in the suspension or generate anti-roll forces to eliminate roll during cornering, and exert anti-pitch forces to control dive during braking and squat during acceleration. Active suspensions minimize the dynamic wheel loads caused from road roughness to improve road holding [1].

1.5 Bond Graph and 20sim Software

To improve the vehicle performance and control the corresponding parameters, it is mandatory to study the dynamics of different vehicle models. The bond graph method is a very convenient approach to study these dynamic systems. It is an explicit graphical tool

that represents the common energy structure of systems, consisting of subsystems connected by power bonds and junctions in an energy conserving way. This bond identifies the power flow path. The basic system elements are represented as resistance, capacitance and inductance, source of effort and source of flow, transformer and gyrator. From the bond graph, the system equations can be derived and algorithmized so that all the efforts and flows on the bond graph can be associated with physical quantities of the physical system [6, 7].

Bond graph method was introduced by late Professor Dr. Henry M. Paynter (1923-2002), MIT and UT Austin, who developed most of the underlying concepts to form a conceptual framework and corresponding notation. It was further studied and elaborated by his students, Dean C. Karnopp and Ronald C. Rosenberg. They designed ENPORT, the first computer tool, that was capable of simulating bond graph models. Later Jan van Dixhoorn, who introduced bond graph method in Europe, realized that a block-diagram-based software TUTSIM could be used to input simple causal bond graphs. His work initiated the development of a port-based computer tool, 20-sim or Twente-sim, at the University of Twente, which extensively uses bond graph method to model and simulate dynamic systems [7]. It allows users to create models intuitively and graphically using block diagrams, physical components as well as using equations. These models can be used to analyze the behaviour of dynamic systems and build control systems with the help of other software packages e.g. 20-sim 4C. 20-sim models can be exported as C-code to 20-

sim 4C which can run them on hardware for rapid prototyping and simulation to control machines and systems in real-time [8].

For this current research, it is convenient to use 20-sim for modeling and simulating an existing quarter car suspension test rig and 20-sim 4C for real-time control of the rig.

1.6 Research Objectives

For the purpose of this thesis, an experimental model of a quarter car with passive suspension is used. The associated sprung and unsprung masses are allowed to slide on a vertical rail. The test rig is used to complete the following research objectives-

1. Different static and dynamic tests need to be done to estimate the values of different parameters and nonlinearities such as suspension stiffness, damping and friction coefficients associated with the setup.
2. Using those values, a passive simulation model for this quarter car setup will be developed in 20-sim to analyze its dynamic responses using bond graph method. The 20sim software will be used to simulate this passive nonlinear quarter car model under various road inputs.
3. The simulation model will be made active by introducing an idealized controller to enhance the performances at sprung and unsprung mass resonances. The actuator gain will be calculated using Riccati equation [2] based on a linear model. The idealized controller performance will be compared for both linear and nonlinear models.

4. A voice-coil actuator is to be installed to make the quarter car setup active. At this stage, 20-sim 4C will be used to import 20-sim model as C-code. Several open and closed loop control tests will be conducted to test the actuator behaviour and develop a control scheme for the active model using 20-sim 4C. The actuator will be first tested using open-loop control signal to verify its expected responses as per its specifications. Then, simple closed-loop position and force control tests will be done to investigate its dynamic responses based on some PID control signals in real-time.
5. The ideal control simulation model will be enhanced by introducing the actuator dynamics to make it realistic. The performance of the both models will be tested and compared.

These works develop the necessary preliminary steps in an overall research program to design a control scheme for the actuator, use it in the test rig to make it active using modified actuator gains based on Riccati equation and assess the performance of the active suspension system compared to that of the passive one.

1.7 Thesis Outline

The thesis paper is organized as follows. The next chapter gives a brief literature review on scaled vehicle models, parameters estimation, optimal control and controller design. Chapter 3 describes the test setup used and different experimental methods used to estimate its parameters. These parameter values are then used to develop a passive

suspension simulation model in 20-sim using the bond graph method. In chapter 4, an LQR (Linear Quadratic Regulator) based active suspension controller is developed using Riccati equation based on a linear model. This idealized controller is then implemented in the passive simulation model to make it active and its performance is assessed with respect to the passive model as well as the linear model.

On the hardware side, a linear motor is introduced in the test rig to represent an active suspension. The 20sim4C software, and associated computer boards, are integrated with the 20sim bond graph simulation software to allow communication of a) control signals from software to hardware, and b) sensor signals from hardware to software. Different open loop and closed loop control tests are conducted to observe and validate the behaviour of the motor and of the suspension deflection sensor, as described in chapter 5. Chapter 6 returns to simulation, making the active model more realistic by introducing actuator dynamics. Actuator dynamics will introduce time delays that will impact future active suspension implementations in hardware. Chapter 6 also compares the performance of both ideal and realistic active control models. Conclusion and potential future works are discussed in chapter 7.

Chapter 2: Literature Review

A relatively high cost is involved with the instrumentation of a full-scale car setup, hence it is highly risky, in terms of money, to test it at its operational limits. Also running a full-scale vehicle test at the limits is not cost effective as well as highly risky in relation to human life. Hence, conducting tests using scaled vehicle models becomes more and more important.

2.1 Scaled Vehicle and Quarter Car

Brennan and Alleyne [9, 10] mention four major motivational factors relating to the use of scaled vehicles: cost, safety, convenience and flexibility. They introduce a scaled novel vehicle testbed: The Illinois Roadway Simulator (IRS) and identify the parameters for the IRS vehicles. These scaled vehicle dynamics are then compared with that of full-scale vehicles for dynamic similitude which indicates the accuracy of the scaled representation of the actual vehicles. Petersheim and Brennan [11] use the dimensionless formulation to rescale vehicle components, particularly the electric motor and battery subsystems. The simulation results of the scaled components are compared with that of the production

sized components. They conclude that the nonlinear scaling of the vehicle components allows performance comparisons across very large size domains.

A quarter-car test rig can be a cost-effective means for performing a relatively large number of accurate and repeatable tests. Langdon and Southward [12] show that the design and performance of their new quarter-car test rig can be a cost-effective solution and meet the wide range of functional requirements: accommodation of a range of actual suspension components, vehicle roll, aerodynamic forces and weight transfer due to braking and acceleration. They use a Porsche 996 suspension and characterize its dynamics by fitting the parameters of a linear dynamic model to the test response data. Tseng and Hrovat [13] examine a simplified 2-DOF quarter car model to characterize the optimal active suspension and compare it with corresponding passive counterparts. As it negates the gravity effects and the limitations of potentially available actuators, this can be considered as the most ideal case of a 2-DOF quarter car model. They show that the active suspension enhances the performance compared to that of the passive one, even when their RMS performances are similar. Evers et al. [14, 15] use a 4-DOF quarter truck model and propose a novel frequency-domain model validation approach to validate it experimentally. They perform the validation using a set of asynchronous repeated measurement data with low signal-to-noise ratios from a real tractor semi-trailer system. The model represents the front vertical and roll dynamics satisfactorily and can be used for active cabin suspension design. Ahmed et al. [16] design a quarter car for railway vehicle suspension system using SolidWorks motion study tool to analyze its dynamic

characteristics based on selected parameters and track input. They conclude that this analysis can be used to develop a physical suspension test rig.

2.2 Nonlinearities and Parameters Estimation

Identification of system parameters and nonlinearities, present in a quarter car test rig, can be a major issue with the realistic application of the test rig. Different test models have different kinds of nonlinear parameters, such as nonlinear friction force and damping coefficient, Coulomb friction etc., and require different approaches to deal with these parameters identification and estimation. Koch et al. [17] design a quarter car test rig integrating a quadricycle suspension system to study the nonlinear dynamic behaviours of active suspension systems. They use system identification techniques to develop the nonlinear model that matches the behaviours of the test rig. This approach uses a physically motivated gray-box model structure and gradient based parameter optimization. Taskin et al. [18] model a quarter car test rig as a lumped parameter system and use forced vibration method to determine the stiffness and damping parameters of this model. They collect acceleration data of the road input, sprung and unsprung masses using piezoelectric accelerometers to obtain the frequency response functions. Experimental and simulation results are compared to fine-tune the estimated parameters. Sandu et al. [19] experiment with a quarter car test rig equipped with a McPherson strut suspension and characterize its dynamic behaviour. The test rig is modelled as both: linear and nonlinear systems. The experimental results are then compared with the responses

of both models. They show that the nonlinear model performs slightly better than the linear one. The University of Iowa's Center for Computer-Aided Design develops a general purpose, multibody dynamics program, Real Time Recursive Dynamics (RTRD). Salaani et al. use this RTRD program to model the rigid body behaviour of suspensions, wheels, steering linkages and chassis of a 1997 Jeep Cherokee model [20, 21] and a 1998 Chevrolet Malibu model [22] for the National Advanced Driving Simulator's vehicle dynamics software, NADSdyna [23]. In this thesis, a combination of forced vibration and free fall tests of masses mounted on linear bearings, representing car body and wheel masses, is done to characterize nonlinear friction for simulation models. The improved accuracy of those simulation models will improve their ability to predict the real performance of controller when they are implemented in hardware.

2.3 Optimal Control

Karnopp [24] shows 'sky-hook' damping, a simple velocity feedback control law, can be used to obtain most of the performance improvement at the sprung mass resonant frequency. According to sky-hook damping theory, the sprung mass is connected to the ground by the spring at one end and to an imaginary straight line (sky-hook) by the damper at the other end to maintain a stable posture. Theoretically, when the damping coefficient reaches an infinite value, the vehicle will not shake. An acceleration sensor mounted on the vehicle can calculate this imaginary line (acceleration = 0) to give feedback to a controller that operates an actuator according to the theory. Verros et. al.

[25] numerically investigate the optimal suspension stiffness and damping parameters of a quarter car model under random road excitation. They first examine the car models with passive damping. Then they investigate the car models with specifically selected suspension damping coefficient so that the model approximates the characteristics of an active suspension system with sky-hook damping. They conclude that semi-active suspension performs better than passive dual-rate dampers. Hrovat [26] surveys different simple, mostly Linear-Quadratic (LQ) based, optimal control concepts on several active suspension models: 1D quarter-car models, 2D half-car models and 3D full-car models, and their practical developments. These optimal control techniques provide a useful understanding about bandwidth requirements, performance potentials and optimal structure of advanced vehicle suspensions.

2.4 Active Control and Controller Design

Lauwerys et al. [27, 28] design and validate experimentally a linear control model approximation of the nonlinear dynamics of an active suspension mounted in a quarter car test rig. This approach is fast, in the sense that it reduces the time-consuming tasks by avoiding the nonlinearities of the model, and well supported by CACSD (Computer Aided Control System Design)-software tools. Its application enhances the performance of an active suspension compared to that of a tuned passive shock absorber and satisfies the results from both simulation as well as experiment. McGinn and Geraghty [29] use state-space methods and the pole placement technique to design a simple and robust active

suspension control system in an unmanned ground vehicle (UGV). This control method can access to all the system states possible for a UGV. They develop continuous and discrete controllers for both 1-DOF and 2-DOF quarter car models and find that active suspension significantly improves the response properties of a 2-DOF suspension by comparing open and closed loop systems.

Chantranuwathana and Peng [30] design a nonlinear active suspension controller for force control loop. The main loop of the controller uses a standard LQ design method to calculate the desired force signal. They propose a modified adaptive robust control method to maintain stability when force sensor is removed from the system. This controller compensates for the hydraulic actuator dynamics to achieve high performance. They show that the proposed controller performs better than a conventional PID controller. Alleyne and Hedrick [31] apply a nonlinear adaptive controller to an electro-hydraulic suspension system and show that it enhances the system performance in terms of improving the ride quality compared to that of the passive system. They use a parameter adaptation scheme, and later modify it to identify the changes in the parameters with the system states, to enhance the controller performance. In this thesis, a simulation model is used to assess the effect of a separate force control loop within the main active suspension control loop. In the idealized scenarios presented in most of the literature on active suspension control, the controller computes a desired actuator force based on system states. The force is assumed to be immediately and exactly available from the actuator. For hardware implementation, the actuator force must be measured,

and the actuator voltage is then modified to change the force to the value commanded by the controller. Therefore, this thesis will simulate a realistic active suspension in which suspension states determine the desired force, the current actual force is measured, and a PID control loop calculates a control voltage for the actuator in order to achieve the desired force as quickly as possible. This will aid in predicting the practically attainable control benefits in future stages of the hardware development.

2.5 Bond Graph Based Vehicle Models

Rideout and Haq [32] illustrate an active modeling technique using quarter- and half-car models subjected to varying road profile. The purpose of this method is to continuously monitor the contribution of an element to overall system dynamics. Bond graph based switch eliminates the element when the moving average value of its absolute power falls below a threshold, but allows its importance to be calculated and it to be 'on' if necessary. Rideout [33] shows the use of bond graph to simulate longitudinal, pitch and bounce dynamics coupled with transverse frame vibration of a truck. It also allows large rigid body motions through using modal expansion of a free-free beam for frame flexibility.

Margolis and Shim [34] use bond graph to develop a 6-DOF, four wheel and nonlinear vehicle dynamic model. The purpose is to use this model to develop and demonstrate controllers for vehicle safety which includes sensors, signal processing and actuation. Deur et. al. [35] present a dynamic modeling approach of HEV powertrains using bond graph and causality to develop minimum-realization models efficient for various simulation and

analyses. They also derive several variants of the HEV powertrain dynamics model for various complexity of driveline structures. Banerjee, et. al. [36] develop an integrated bond graph model of a railway truck with 18-DOF to eliminate the limitations of various assumptions such as reduced DOF, small displacements and absence of kinematic nonlinearities and inclination of the contact surface. This model is used to simulate the truck stability behavior and curving performance for given track conditions and can be used to determine critical speed and to design track layout for given rail vehicle.

The widespread use of the bond graph proves that it provides a viable simulation environment and confirms its acceptance in the vehicle modeling community. The bond graph method allows to model any system with elements from different energy domains. One can easily identify the explicit system equations, the presence of algebraic loops and dependent states that create implicit equations, only by visually inspecting the bond graph of that system [37]. Because of its simplicity and widespread advantages, the bond graph method is used in this research. As mentioned earlier, the most popular commercial bond graph software can also be connected to hardware (single board computer, DAQ board etc.) with an associated software package from the same company. Therefore, bond graphs will facilitate both the simulation and hardware aspects of this research.

2.6 Literature Motivation

Based on the above literatures, to study and experiment with vehicle suspension system, it is safe and cost effective to use a scaled vehicle model. In this research, a quarter car

test rig is used for that purpose whose suspension parameters and associated nonlinearities are unknown. Different static and dynamic tests are done to estimate the values of these parameters and nonlinearities. These values are then used to develop bond graph based simulation models in 20-sim software for both passive and active models. The active model is exported to 20-sim 4C as C-code to design the controller for the actuator.

The next chapter will discuss the physical model of the existing quarter car test rig and different methods of estimating its parameters. These parameter values will then be used to develop a 20-sim model of a passive quarter car.

Chapter 3: Parameterizing a Simulation Model of a Quarter Car Test Rig

A quarter car represents the vehicle system at each wheel. A simple quarter car with two degrees of freedom consists of a sprung mass and an unsprung mass, suspension and tire spring and damper. The sprung mass (m_s) represents the one-fourth of the vehicle body mass, supported by the suspension spring (k_s) and damper (b_s). The unsprung mass (m_u) below the suspension represents the equivalent mass of the axle and tire. A spring supporting the unsprung mass represents the tire stiffness (k_t) [2]. In this research, a quarter car test rig is used which was developed based on this simple 2-DOF quarter car model, as shown in figure 3.1.

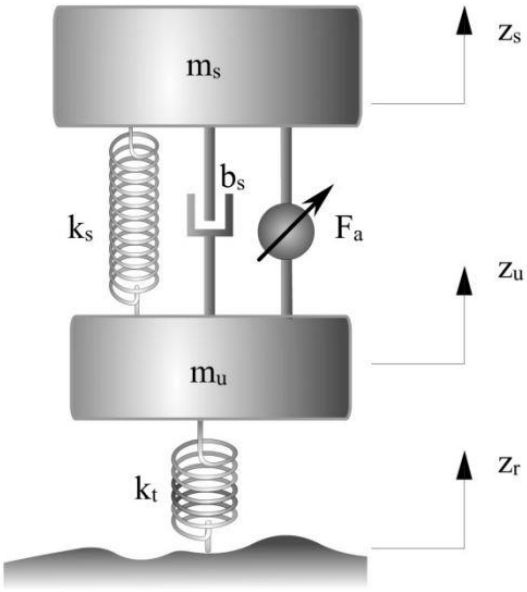


Figure 3.1: Quarter car active suspension model [2].

3.1 Physical Model

An existing quarter car test rig, as shown in figure 3.2, is used. Its different parameters and nonlinearities required to develop the simulation model are then estimated by various static and dynamic tests. The test rig parameters to be estimated are –

- suspension stiffness,
- suspension damping
- tire stiffness, and
- rail friction.

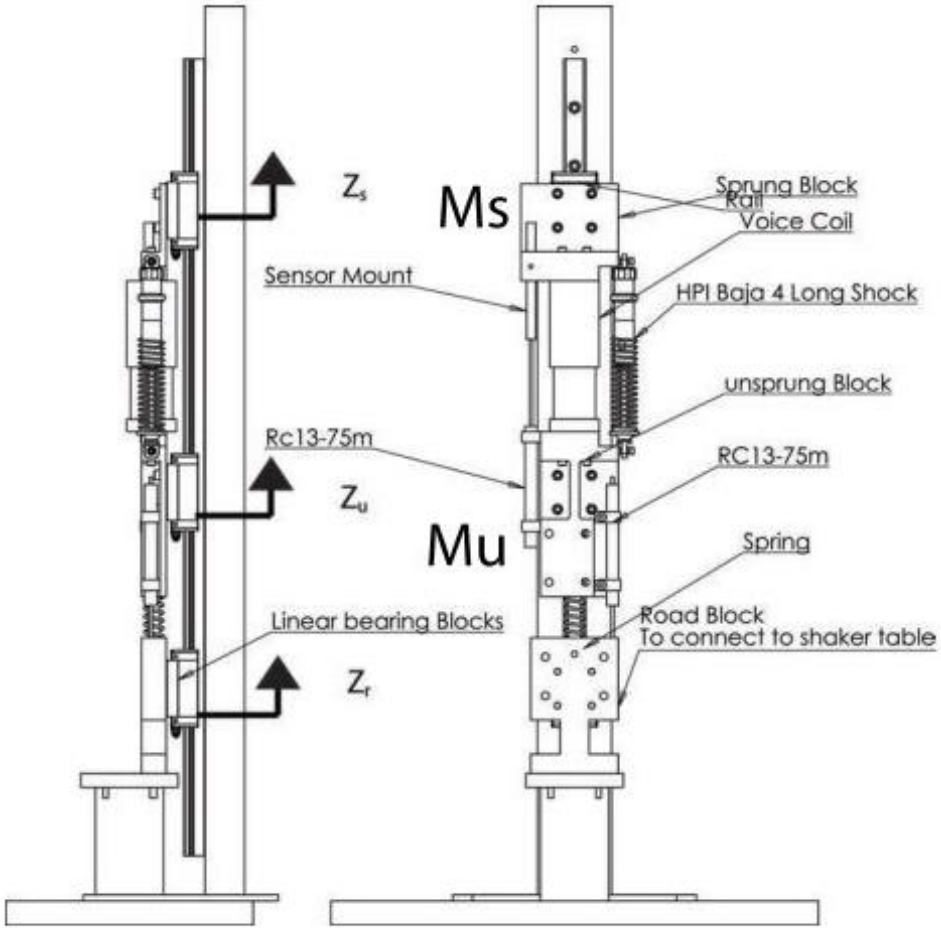


Figure 3.2: Schematic of quarter car test rig [38].

The sprung mass and unsprung mass of the test rig are allowed to slide on a vertical rail. A shock absorber (manufacturer: HPI Racing Ltd., model: HPI Baja 4 Long Shock) which consists of a damper and two springs is placed in between these two masses. The tire is represented by a relatively very stiff spring. There is a metal block below the tire spring which has a provision to connect the test rig to shaker table and allows the test rig to experience the road profile.

The setup is equipped with 2 LPDTs (model RC13-75M): one in between the sprung and unsprung masses to measure z_s and the other in between the unsprung mass and the road block to measure z_u, and 2 accelerometers (model 1221L-002), one on each mass. A voice coil actuator (model LVCM-051-089-01) is placed in between the sprung and unsprung masses to make the model active by providing active force (F_a). The actuator is either removed or remains inactive during the experiments of passive suspension model. A load cell (model LCCA-250) is used to measure dynamic forces during different experiments.

3.2 Spring Stiffness Estimation

Simple static compressive load tests are done (figure 3.3) and the different spring stiffnesses are obtained from load vs. deflection curves. It is observed that the springs are very linear in nature, as seen in figure 3.4.

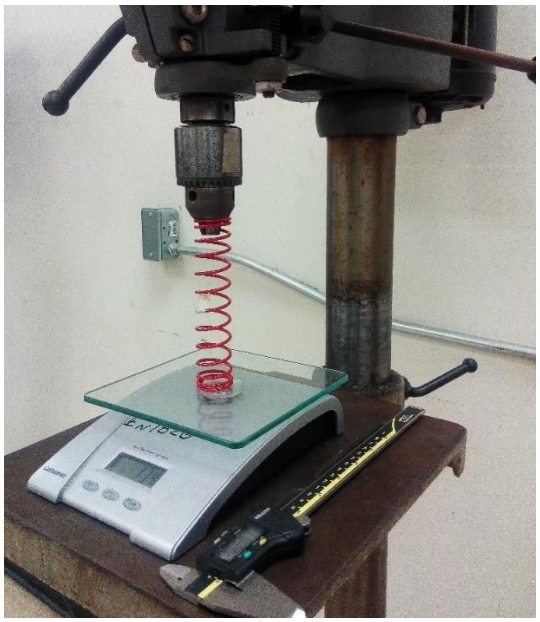


Figure 3.3: Spring test.

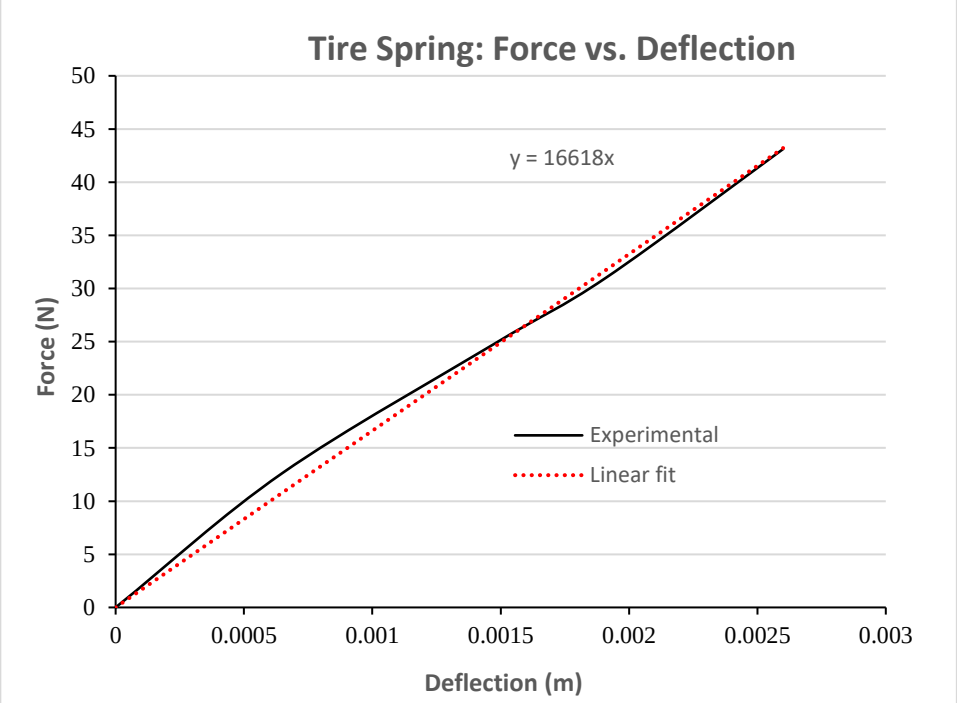


Figure 3.4: Tire spring stiffness.

The complete data and plots related to suspension and tire spring tests are given in appendix A. In the shock absorber, two springs are placed in series with a small plastic slider in between them. Therefore, the suspension stiffness (k_s) can be calculated using the relationship shown below,

$$\frac{1}{k_s} = \frac{1}{k_1} + \frac{1}{k_2} \tag{3.1}$$

Table 3.1: Quarter car model specifications.

Parameters	**Values**
Sprung mass (m_s)	7.088 kg
Unsprung mass (m_u)	2.305 kg
Suspension stiffness (k_s)	1157.25 N/m
Tire stiffness (k_t)	16618 N/m

3.3 Rail Friction and Suspension Damping Estimation

Several simple dynamic tests are conducted to measure the friction and damping associated with the rail and shock absorber. All these experimental data are analyzed using 20sim.

First, only the unsprung mass is cycled along the rail in the vertical position of the test rig. The dynamic force (F) is applied on the load cell, from a static equilibrium position of the unsprung mass on a spring (k), as shown in figure 3.5. An accelerometer, an LPDT and the load cell capture its acceleration, displacement and total dynamic force respectively. The cyclic change in the dynamic force indicates the sum of the inertial and friction force of the unsprung mass motion. The inertial force of the unsprung mass, calculated using the acceleration data, is subtracted from that cyclic change in the measured force to obtain the friction force between the rail and the unsprung mass, as shown in figure 3.6.

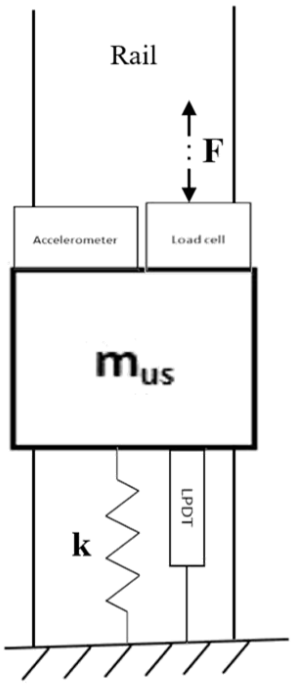


Figure 3.5: Rail friction test with unsprung mass.

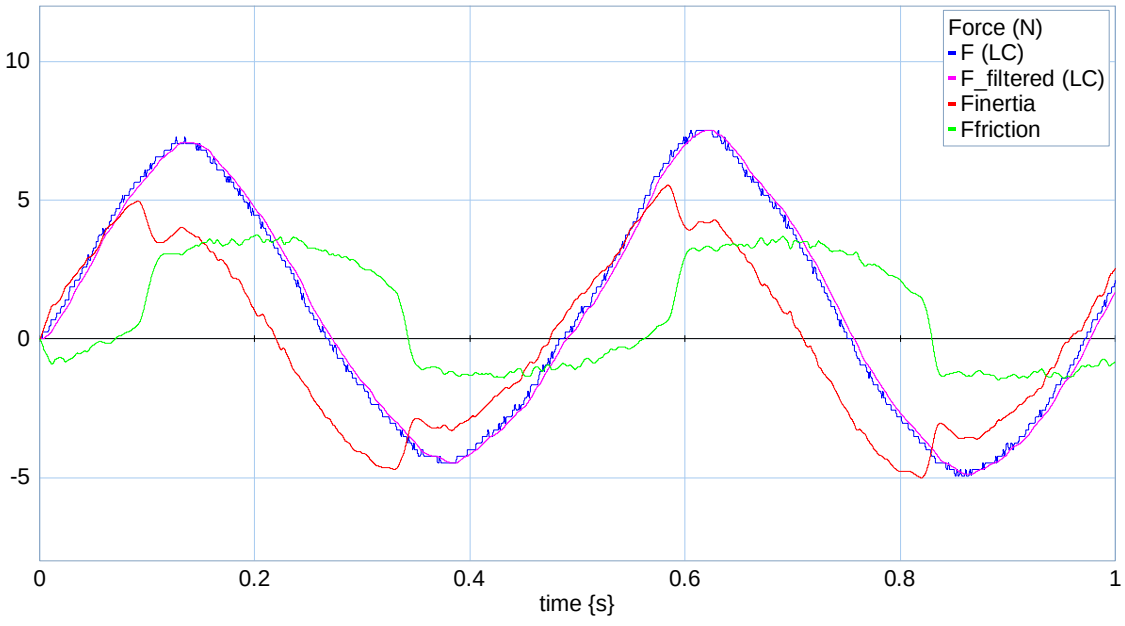


Figure 3.6: Forces during unsprung mass cyclic motion.

This friction force is then plotted against the unsprung mass velocity, calculated using the LPDT data. This plot indicates that a Coulomb friction of about 3 N is present in the rail when the unsprung mass is in motion, as can be seen in figure 3.7.

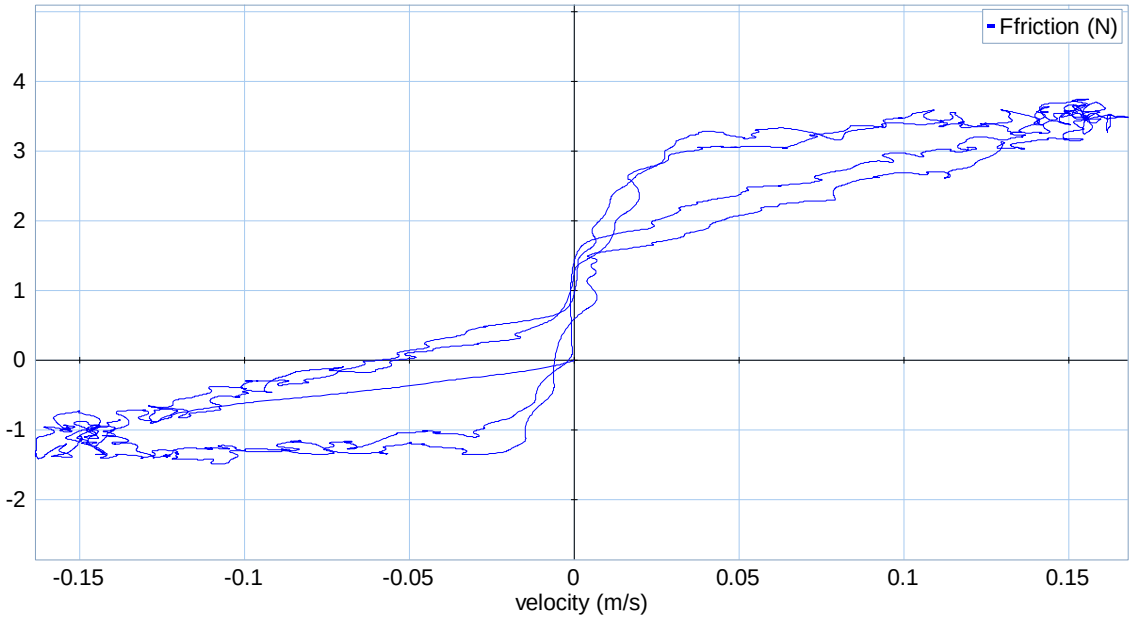


Figure 3.7: Friction force vs. unsprung mass velocity.

Again, the unsprung mass motion is tested, using an accelerometer, by allowing it to fall freely along the rail. A simple bond graph model (figure 3.8) is developed to match this motion under gravity with the simulated motion. By comparing the test and simulation results, it is realized that Coulomb and drag frictions are associated with the rail. By trial and error, a Coulomb friction of 5 N and a drag friction coefficient of 1.25 Ns^2/m^2 predict the test result well. Therefore, a Coulomb friction of about 3-5 N is associated with the unsprung mass motion along the rail. Sigmoidal function is used to remove the

discontinuity, in the friction function, due to the Coulomb friction. Now, the rail friction for unsprung mass motion (assuming a Coulomb friction of 5 N) can be expressed as,

$$F_{f1} = 5 * \left(\frac{2}{1 + e^{-10*v_u}} - 1\right) + 1.25 * v_u^2 * sign(v_u) \qquad (3.2)$$

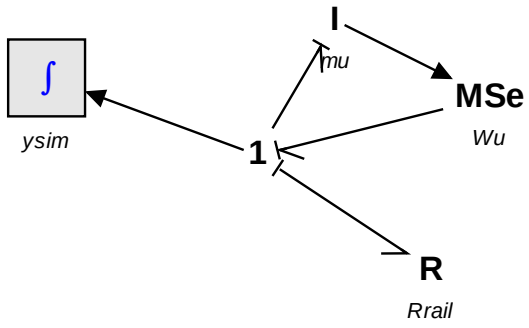


Figure 3.8: Unsprung mass free fall model.

The comparison between the experimental result and the simulation output using equation 3.2 shows that the terminal velocity is well predicted, as shown in figure 3.9.

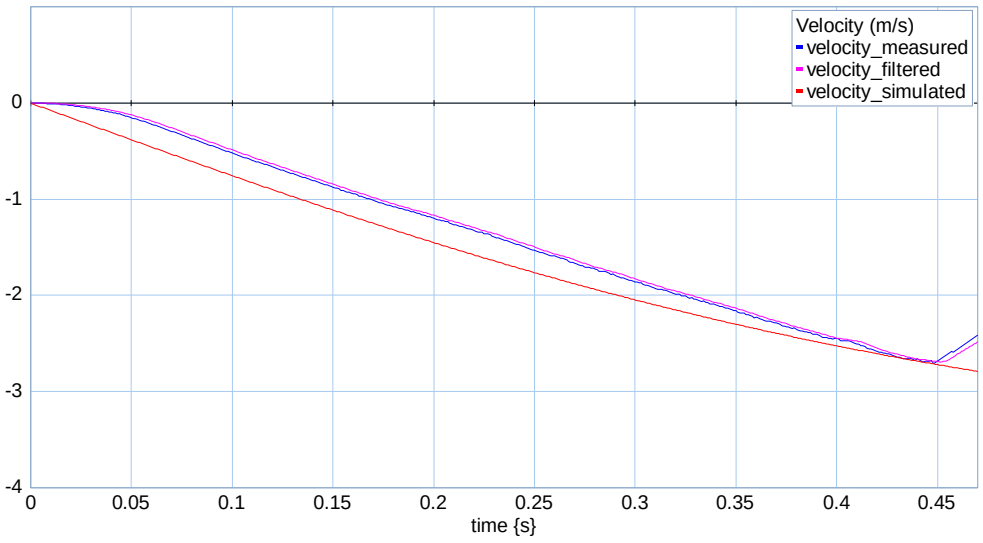


Figure 3.9: Comparison of the experimental and the simulated unsprung mass free fall.

In another test, the sprung mass is slowly pressed against the suspension with a load cell on the other end, in the horizontal position of the test rig, as shown in figure 3.10, to measure the static friction, i.e., the minimum force that is required to start moving the sprung mass on the rail. The measured force by the load cell is then plotted against the deflection of the suspension, measured by an LPDT. This plot shows a static friction force of about 8 N for the sprung mass motion with a negligible displacement, as seen in figure 3.11. Since the sprung mass is heavier than the unsprung mass, it is expected that the static friction is higher in this case. Also, the slope of this plot is supposed to be equal to the value of suspension stiffness of 1157.25 N/m obtained from the spring test. From the plot, the value of the slope is found to be about 1121 N/m which validates both experiments. Now, the rail friction for sprung mass motion can be approximated using sigmoidal function as,

$$F_{f2} = 8 * \left(\frac{2}{1 + e^{-10*v_s}} - 1\right) + 1.25 * v_s^2 * sign(v_s) \qquad (3.3)$$

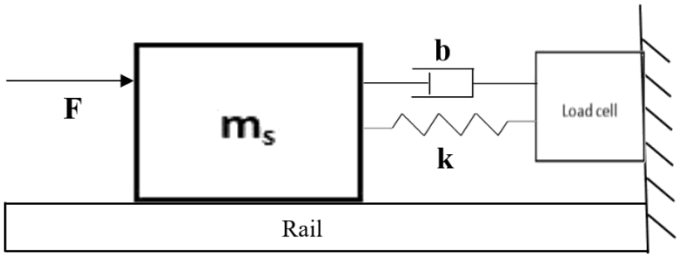


Figure 3.10: Static rail friction test with sprung mass.

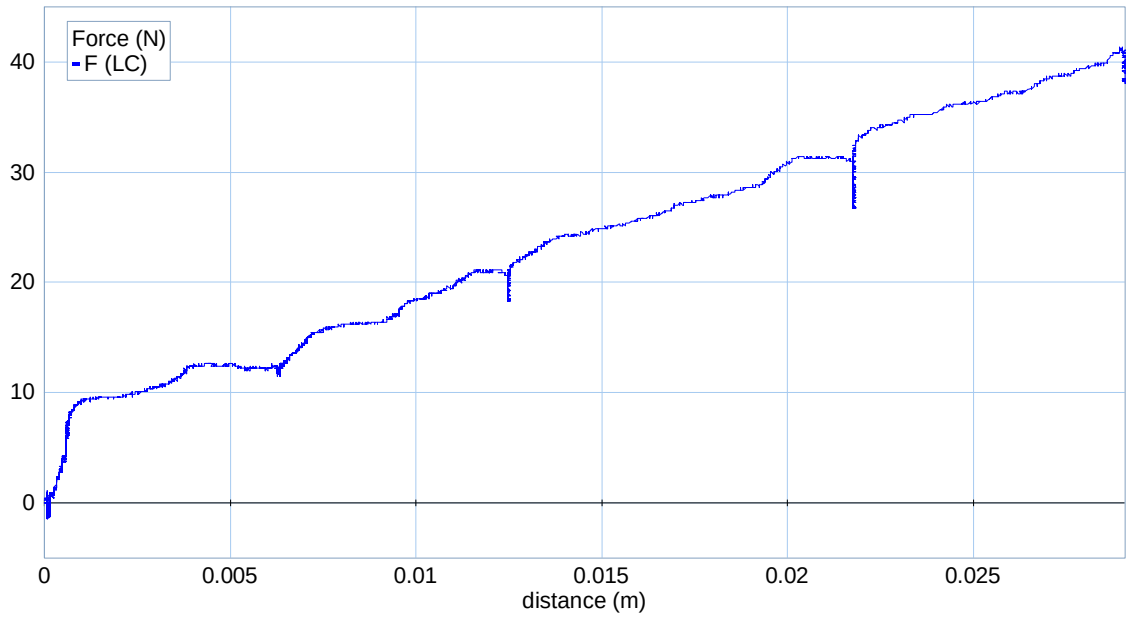


Figure 3.11: Force vs. suspension deflection for sprung mass motion.

Another cyclic forced vibration test is conducted using the sprung mass to calculate the damping coefficient values of the shock absorber. The sprung mass is cycled against the shock absorber while the dynamic force is applied on the load cell placed on the sprung mass, as shown in figure 3.12. The load cell reading and the inertial force are subtracted from the suspension spring force to obtain the damping force. This damping force is then plotted against the suspension velocity, as shown in figure 3.13.

For a linear viscous damper, the slope of this curve would define the damping coefficient of the suspension. But here the curve is nonlinear and the shape of the curve is very much comparable to that of some sigmoidal function. Thus, a sigmoidal function is fitted to the curve through trial and error to capture the damping force-velocity relation properly,

neglecting the hysteresis effect. So, the suspension damping force can be approximated as,

$$F_d = 10.5 * \left(\frac{2}{1 + e^{-100 * v_{susp}}} - 1\right) + 25 * v_{susp} \quad (3.4)$$

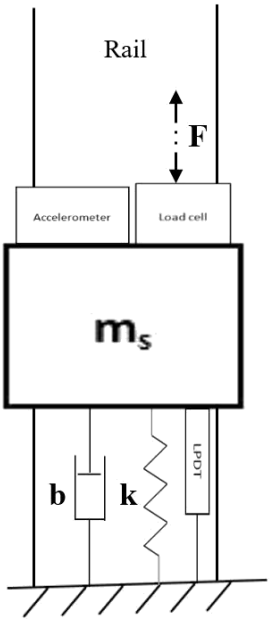


Figure 3.12: Suspension damping force test with sprung mass.

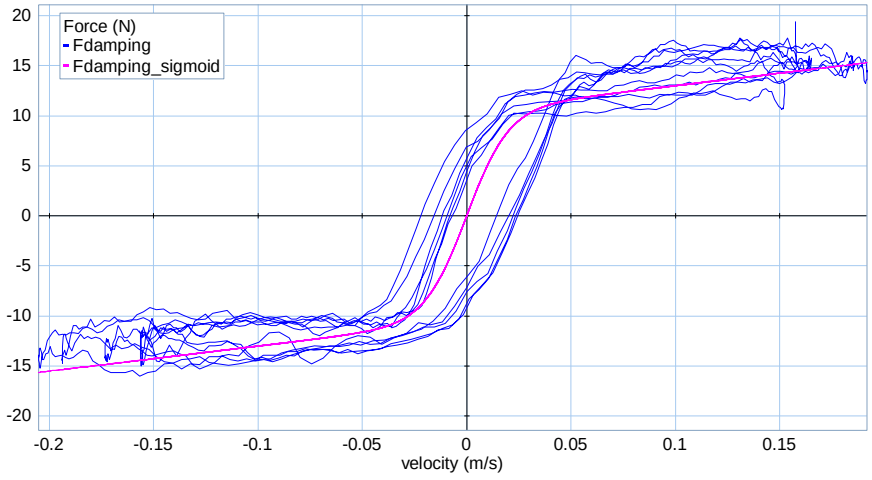


Figure 3.13: Damping force vs. suspension velocity.

3.4 20-sim Model for Passive Quarter Car

Using the experimentally determined test rig parameters discussed before, a bond graph model in 20-sim is developed for the 2-DOF passive quarter car, as shown in figure 3.14. The modified 20-sim codes for the bond graph elements are included in appendix B.

A bond graph conserves the energy of the system. Generalized inertias (I) and capacitances (C) store energy while generalized resistors (R) dissipate energy from the system, as a function of the system state variables. The state variables are generalized momentum (p) and displacement (q), and their time derivatives are generalized effort (e) and flow (f) respectively. The product of generalized effort and flow is generalized power. The power flows among different elements and their paths can be identified from the bond graph. The system interacts with its surroundings through sources of effort (Se) and flow (Sf) [39].

Energy is transferred and/or converted among different elements through junction elements, such as, power-continuous generalized transformers (TF) and gyrators (GY), and 1- and 0-junctions. They conserve power during the energy transfer. The ratio of this exchange is a constant or a function of external variables in case of modulated transformers (MTF) and gyrators (MGY). 1- and 0-junctions satisfy Kirchoff's loop and node laws. Net effort is zero around a 1-junction and it maintains a common flow in the connected elements. Net flow is zero around a 0-junction and it maintains a common effort in the connected elements [39].

A full arrow indicates a powerless information flow as a modulating signal. A half-arrow power bond represents algebraically positive power flow direction. The output or input (either effort or flow) of the connected element can be identified from the position of the causal stroke normal to its bond [39]. Table 3.2 shows the symbols and the constitutive laws based on the causal strokes for the different bond graph elements.

Table 3.2: Bond graph elements [39].

	SYMBOL	CONSTITUTIVE LAW (LINEAR)	CAUSALITY CONSTRAINTS
SOURCES			
Flow	Sf ⊢→	f = f(t)	fixed flow out
Effort	Se →⊣	e = e(t)	fixed effort out
ENERGETIC ELEMENTS			
Inertia	→⊣ I ⊢→ I	$f = \frac{1}{I}\int e\,dt$ $e = I\frac{df}{dt}$	preferred integral
Capacitor	⊢→ C →⊣ C	$e = \frac{1}{C}\int f\,dt$ $f = C\frac{de}{dt}$	preferred integral
Resistor	⊢→ R →⊣ R	$e = Rf$ $f = \frac{1}{R}e$	none
2-PORT ELEMENTS			
Transformer	1 →⊣ TF ⊢→ 2 (n)	$e_2 = n\,e_1$ $f_1 = n\,f_2$	effort in-effort out or flow in-flow out
Modulated Transformer	θ ↓ MTF n(θ)	$e_2 = n(\theta)\,e_1$ $f_1 = n(\theta)\,f_2$	
Gyrator	1 ⊢→ GY ⊢→ 2 (n)	$e_2 = n\,f_1$ $e_1 = n\,f_2$	flow in-effort out or effort in-flow out
Modulated Gyrator	θ ↓ MGY n(θ)	$e_2 = n(\theta)\,f_1$ $e_1 = n(\theta)\,f_2$	
CONSTRAINT NODES			
1-junction	1 →⊣ 1 ⊢→ 2 ↓3	$e_2 = e_1 - e_3$ $f_1 = f_2$ $f_3 = f_2$	one flow input
0-junction	1 ⊢→ 0 ⊢→ 2 ↓3	$f_2 = f_1 - f_3$ $e_1 = e_2$ $e_3 = e_2$	one effort input

Similarly, in figure 3.14, a road profile velocity signal (vr_signal) is taken into a modulated flow source (vr_source) as a powerless flow. This MSf supplies this flow to the tire. The tire capacitance (kt) stores some energy and creates a velocity difference between the road and the unsprung mass.

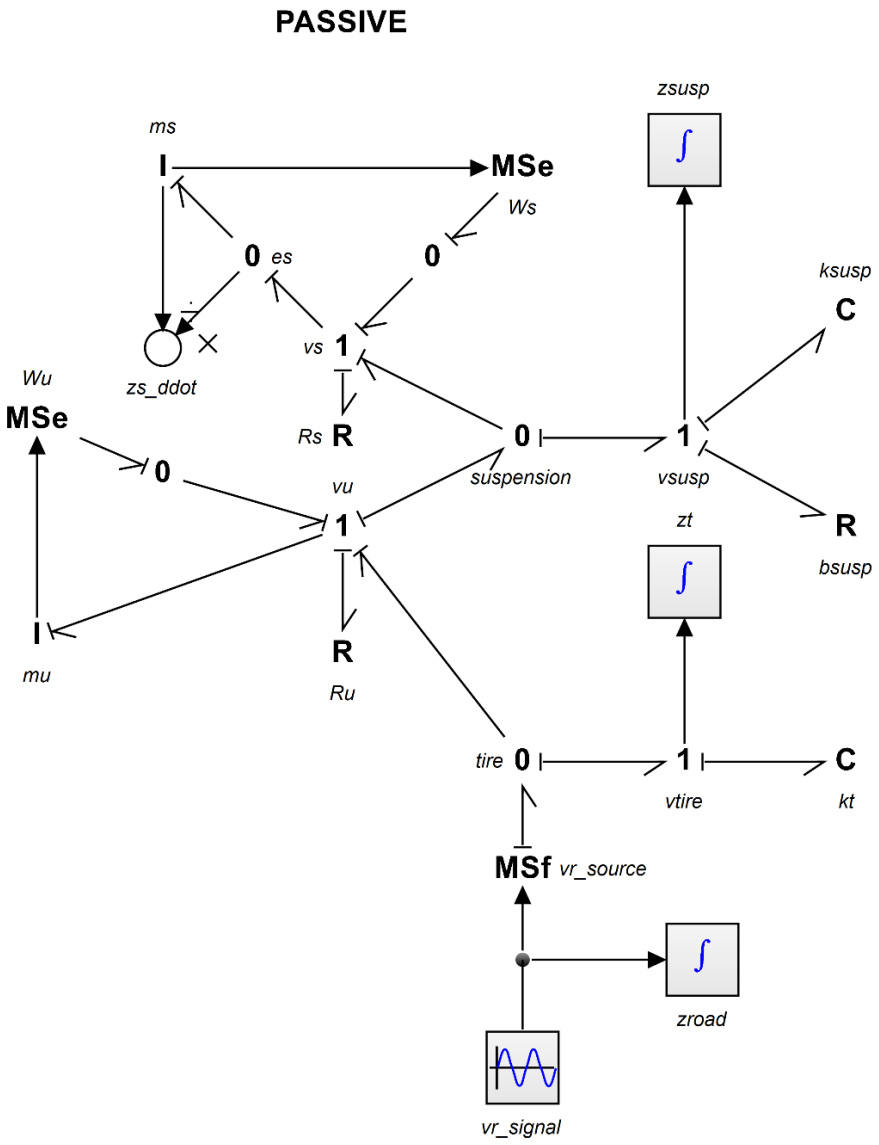


Figure 3.14: Bond graph model for 2-DOF passive quarter car.

The rail friction (Ru) dissipates some energy due to the unsprung mass motion. The unsprung mass (mu) shows some inertial effect and includes some effort into the system due to its weight (Wu). Then the suspension capacitance (ksusp) stores some energy while the suspension damper (bsusp) dissipates some energy, and creates a velocity difference between the unsprung and sprung mass. Again, the rail friction (Rs) dissipates some energy due to the sprung mass motion. The sprung mass (ms) has some inertial effect on the system and includes some effort into the system due to its weight (Ws).

3.5 Equations of Motion for the Linearized Model

The quarter car system equations of motion for a linear model, in standard second order matrix form, can be represented as,

$$M\ddot{z} + C\dot{z} + Kz = H_1 z_r + H_2 \dot{z}_r + H_3 F_a + H_4 g \tag{3.5}$$

Where,

$$M = \begin{bmatrix} m_s & 0 \\ 0 & m_u \end{bmatrix}$$

$$C = \begin{bmatrix} b_s & -b_s \\ -b_s & b_s + b_t \end{bmatrix} \tag{3.6}$$

$$K = \begin{bmatrix} k_s & -k_s \\ -k_s & k_s + k_t \end{bmatrix}$$

$$z = [z_s \quad z_u]^T \tag{3.7}$$

and,

$$H_1 = [0 \quad k_t]^T$$
$$H_2 = [0 \quad b_t]^T$$
$$H_3 = [1 \quad -1]^T$$
$$H_4 = [m_s \quad m_u]^T \tag{3.8}$$

From the bond graph of the model in figure 3.14, the state variables are easily identifiable. The general linear system of the 2-DOF quarter car model considering gravitational effect can be described by,

$$\dot{x} = Ax + BF_a + Dg + L\dot{z}_r \tag{3.9}$$

Where, state variable matrix,

$$x = \begin{bmatrix} z_{susp} & \dot{z}_s & z_t & \dot{z}_u \end{bmatrix}^T \tag{3.10}$$

Knowing the state variables, the state equations for the system can be developed from the bond graph using the fact that the derivatives of momentum and displacement are equal to the effort and the flow respectively. Assuming the system to be linear, i.e., the rail friction and the suspension damping are linear, the state equations are found as follows,

$$\dot{z}_{susp} = \dot{z}_u - \dot{z}_s \tag{3.11}$$

$$\ddot{z}_s = \frac{k_s}{m_s} * z_{susp} - \left(\frac{b_s + b_1}{m_s}\right) * \dot{z}_s + \frac{b_s}{m_s} * \dot{z}_u - \frac{1}{m_s} * F_a - g \tag{3.12}$$

$$\dot{z}_t = \dot{z}_r - \dot{z}_u \tag{3.13}$$

$$\ddot{z}_u = \frac{b_s}{m_u} * \dot{z}_s - \frac{k_s}{m_u} * z_{susp} - \left(\frac{b_s + b_2}{m_u}\right) * \dot{z}_u + \frac{k_t}{m_u} * z_t + \frac{1}{m_u} * F_a - g \tag{3.14}$$

Comparing these state equations 3.11 – 3.14 with equations 3.9 and 3.10, matrices A, B, D and L for the linear system can be found as follows,

$$A = \begin{bmatrix} 0 & 1 & 0 & -1 \\ \frac{-k_s}{m_s} & \frac{-(b_s + b_1)}{m_s} & 0 & \frac{b_s}{m_s} \\ 0 & 0 & 0 & 1 \\ \frac{k_s}{m_u} & \frac{b_s}{m_u} & \frac{-k_t}{m_u} & \frac{-(b_s + b_2)}{m_u} \end{bmatrix} \tag{3.15}$$

$$B = \begin{bmatrix} 0 & \frac{1}{m_s} & 0 & \frac{-1}{m_u} \end{bmatrix}^T \tag{3.16}$$

$$D = [0 \quad 1 \quad 0 \quad 1]^T \tag{3.17}$$

$$L = [0 \quad 0 \quad -1 \quad 0]^T \tag{3.18}$$

The next chapter will discuss the design and development of an LQR (Linear Quadratic Regulator) based active suspension controller for this nonlinear test rig model. The controller will be designed based on a linear model and then it will be applied to the nonlinear and that linear models. The comparative result will give some idea about the difference between its performances in real (nonlinear) and ideal (linear) situations. The attenuation of acceleration, rattle space and tire deflection transfer functions will be studied, using 20-sim models of this passive system and an active suspension system based on this controller, to judge the effectiveness of the active suspension system. The acceleration, rattle space and tire deflection transfer functions are shown in equations 3.19 – 3.21 respectively [2],

$$H_A(s) = \frac{\ddot{z}_s(s)}{\dot{z}_r(s)} \tag{3.19}$$

$$H_{RS}(s) = \frac{z_s(s) - z_u(s)}{\dot{z}_r(s)} \tag{3.20}$$

$$H_{TD}(s) = \frac{z_u(s) - z_r(s)}{\dot{z}_r(s)} \tag{3.21}$$

Chapter 4: Design and Simulation of an Active Suspension Controller

The influence of specific suspension parameters on the suspension performance can be investigated by calculating the magnitude of the suspension transfer functions. It is seen that reduced suspension stiffness improves sprung mass response at high frequencies but increases suspension deflection at low frequencies. Increased suspension damping reduces or eliminates the first resonant peak of the sprung mass and the suspension deflection but incorporates high frequency 'harshness'. Increased tire stiffness improves tire deflection at low frequencies but increases the second resonant peak [2].

In this chapter, an active suspension controller will be designed based on a linear model assumption and then applied to the nonlinear test rig model. The controller performance will be compared using both the nonlinear and that linear models, to show the effect of real-world nonlinearities on the idealized controller performance.

4.1 Modal Decoupling, Natural Frequencies and Mode Shapes

Assuming the system to be linear and undamped, and no force is applied, the natural frequencies ω_i of this system can be represented as,

$$\det(-\omega_i^2 M + K) = 0 \tag{4.1}$$

Where, i = 1, 2, …, n.

Now, the mode shapes ϕ_i are chosen such that,

$$[-\omega_i^2 M + K]\phi_i = 0 \tag{4.2}$$

And, the mode shapes can be mass-normalized so that the mass-normalized modal matrix P satisfies,

$$P^T M P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I \tag{4.3}$$

$$P^T K P = \Lambda \tag{4.4}$$

Where, $P = [\phi_1\ \phi_2 \ldots\ \phi_n]$ and $\Lambda = \begin{bmatrix} \omega_1^2 & 0 & \ldots & 0 \\ 0 & \omega_2^2 & 0 & 0 \\ \vdots & 0 & \ddots & \vdots \\ 0 & 0 & \ldots & \omega_n^2 \end{bmatrix}$.

Now, the coordinates can be changed accordingly,

$$r = P^T M z \tag{4.5}$$

It decouples the equation 3.5 in new coordinates of the equation 4.5,

$$\ddot{r} + P^T C P \dot{r} + \Lambda r = P^T H_1 z_r + P^T H_2 \dot{z}_r + P^T H_3 F_a + P^T H_4 g \qquad (4.6)$$

Using the approximation $k_t >> k_s$, simplification of equation 4.1 would give the equations for two undamped natural frequencies ω_1 and ω_2 as,

$$\omega_1 = \sqrt{\frac{k_s}{m_s}} \qquad (4.7)$$

$$\omega_2 = \sqrt{\frac{k_t}{m_u}} \qquad (4.8)$$

For the parameters of the experimental setup, the approximate natural frequencies would be ω_1=12.78 rad/s and ω_2 = 85.09 rad/s. Using Matlab, the exact values of the system natural frequencies are 12.35 rad/s and 88.05 rad/s which are related to sprung and unsprung masses respectively. So, it is expected that the Bode plots of the equations 3.19 – 3.21 will show sprung mass and unsprung mass resonances at about 12 rad/s and 88 rad/s respectively.

Now, solving equation 4.3 using Matlab, the mass-normalized modal matrix P for the test rig is found to be,

$$P = \begin{bmatrix} 0.3756 & -0.0006 \\ 0.0011 & 0.6587 \end{bmatrix} \quad (4.9)$$

Using equations 4.4 and 4.9, Λ is found to be,

$$\Lambda = \begin{bmatrix} 12.74^2 & 0 \\ 0 & 88^2 \end{bmatrix} \approx \begin{bmatrix} \omega_1^2 & 0 \\ 0 & \omega_2^2 \end{bmatrix} \quad (4.10)$$

The Matlab code used to solve the above equations and find the values of modes and natural frequencies of the test rig is given in appendix C.

4.2 LQR Based Controller Design

The objective of a controller design is to maintain an ideal state of the system with some method of actuation. The method used follows Butsuen's formulation [40]. The performance index J is weighted using factors p_1, p_2, p_3 and p_4 which emphasize suspension deflection, ride quality, tire deflection and unsprung mass velocity respectively and defined as the following quadratic form,

$$J = \int_0^\infty [\ddot{z}_s^2 + p_1(z_s - z_u)^2 + p_2\dot{z}_s^2 + p_3(z_u - z_r)^2 + p_4\dot{z}_u^2]\, dt \quad (4.11)$$

Assuming the system to be linear, the performance index J can be rewritten in the following matrix form,

$$J = \int_0^\infty (x^T Q x + 2x^T N u + u^T R u)\, dt \tag{4.12}$$

Where,

$$Q = \begin{bmatrix} \frac{k_s^2}{m_s^2} + p_1 & \frac{b_s k_s}{m_s^2} & 0 & -\frac{b_s k_s}{m_s^2} \\ \frac{b_s k_s}{m_s^2} & \frac{b_s^2}{m_s^2} + p_2 & 0 & -\frac{b_s^2}{m_s^2} \\ 0 & 0 & p_3 & 0 \\ -\frac{b_s k_s}{m_s^2} & -\frac{b_s^2}{m_s^2} & 0 & \frac{b_s^2}{m_s^2} + p_4 \end{bmatrix}$$

$$N = \begin{bmatrix} -\frac{k_s}{m_s^2} \\ -\frac{b_s}{m_s^2} \\ 0 \\ \frac{b_s}{m_s^2} \end{bmatrix} \tag{4.13}$$

$$R = \frac{1}{m_s^2}$$

The state feedback law is,

$$F_a = -Gx \tag{4.14}$$

which minimizes the performance index and where, G= feedback gain. The feedback gain G can be found by solving the Riccati equation given below,

$$(A - BR^{-1}N)^T\bar{P} + \bar{P}(A - BR^{-1}N) + (Q - N^TR^{-1}N) - \bar{P}BR^{-1}B^T\bar{P} = 0 \quad (4.15)$$

$$G = R^{-1}(B^T\bar{P} + N) \quad (4.16)$$

Where, $\bar{P}$ is given by the positive semi-definite solution to the Riccati equation.

In this formulation, the coefficient matrix D for gravitational effect and the non-linear rail friction are not considered, and the suspension damping coefficient is approximated to be linear with a value of 25 Ns/m, which is the same as that for higher velocities.

4.3 20-sim Model for Active Quarter Car

The passive model is now modified to make it active by introducing the controller in the model, as shown in figure 4.1. A modulated effort source (MSe) is used in the 20-sim model to represent the voice coil actuator. It takes the four states (suspension deflection, sprung mass velocity, tire deflection and unsprung mass velocity) as a single column matrix input which is then multiplied by the feedback gain G to obtain the desired force as output.

To calculate the feedback gain G for different ride conditions, using the Riccati equation according to the above discussion, the values of the weighting factors p_1 through p_4 are chosen accordingly. The sprung mass acceleration can be heavily penalized using very

small values for the weighting factors (p_1 = 0.40, p_2 = 0.16, p_3 = 0.40 and p_4 = 0.16) [2]. For these values, the value of feedback gain G found using Matlab is,

$$G = [-1152 \quad -21 \quad 5 \quad 24.5] \tag{4.17}$$

Ride quality can be moderately weighted using higher values for the weighting factors (p_1 = 400, p_2 = 16, p_3 = 400 and p_4 = 16) [40]. In this case, the calculated feedback gain G is,

$$G = [-992.39 \quad 27.42 \quad 8.77 \quad 0.51] \tag{4.18}$$

The Matlab code used to calculate the feedback gain G is given in the appendix C.

4.4 Controller Performance

4.4.1 Performance Comparison Based on Linear and Nonlinear Models

Since the controller is designed based on a linear model, it is appropriate to compare its performance using the linear model and nonlinear model of the test rig. A road profile equivalent to a 72 km/h car passing a 30 cm long 2 cm high road bump is used, as shown in figures 4.2 and 4.3.

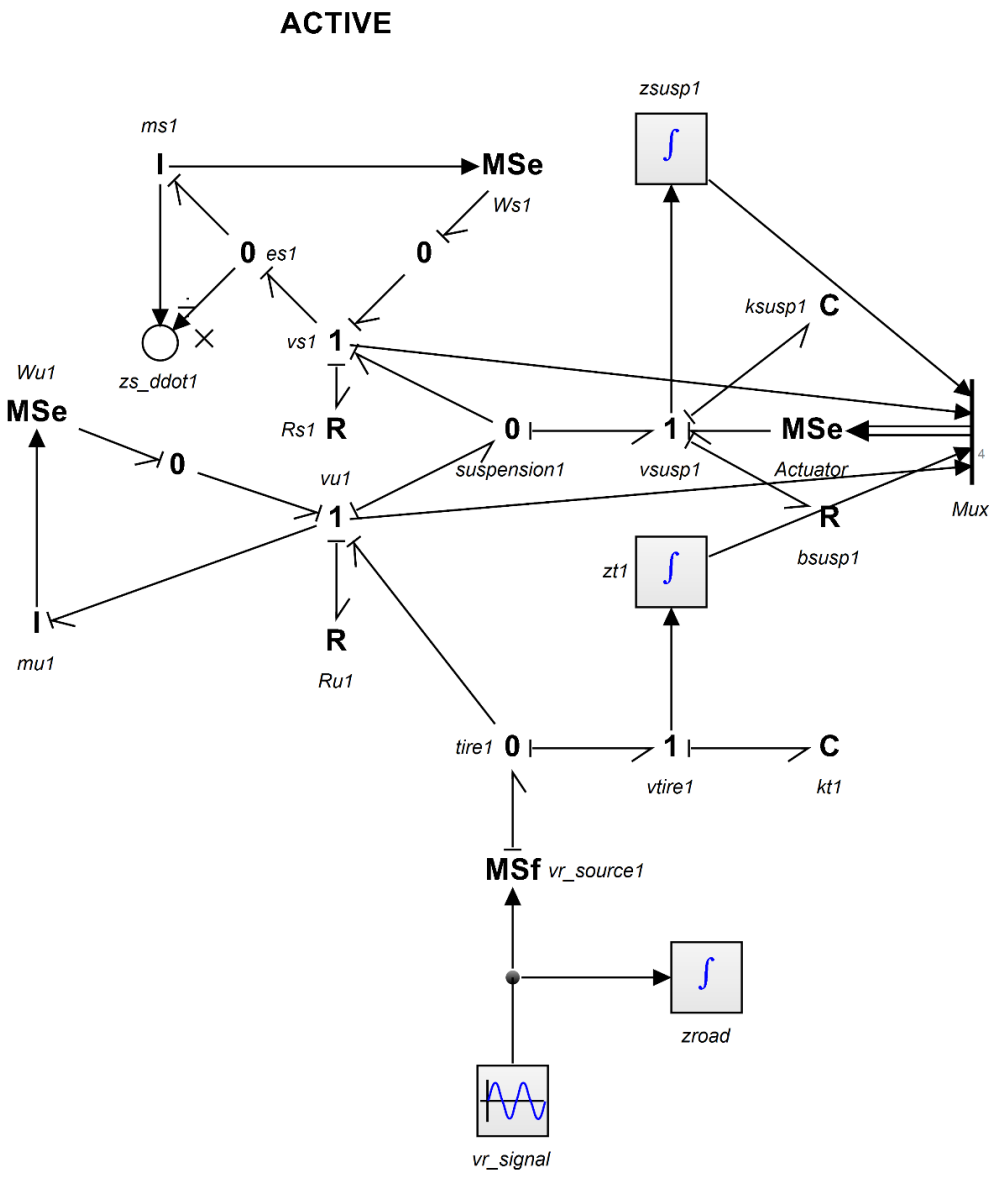


Figure 4.1: Bond graph model for 2-DOF active quarter car.

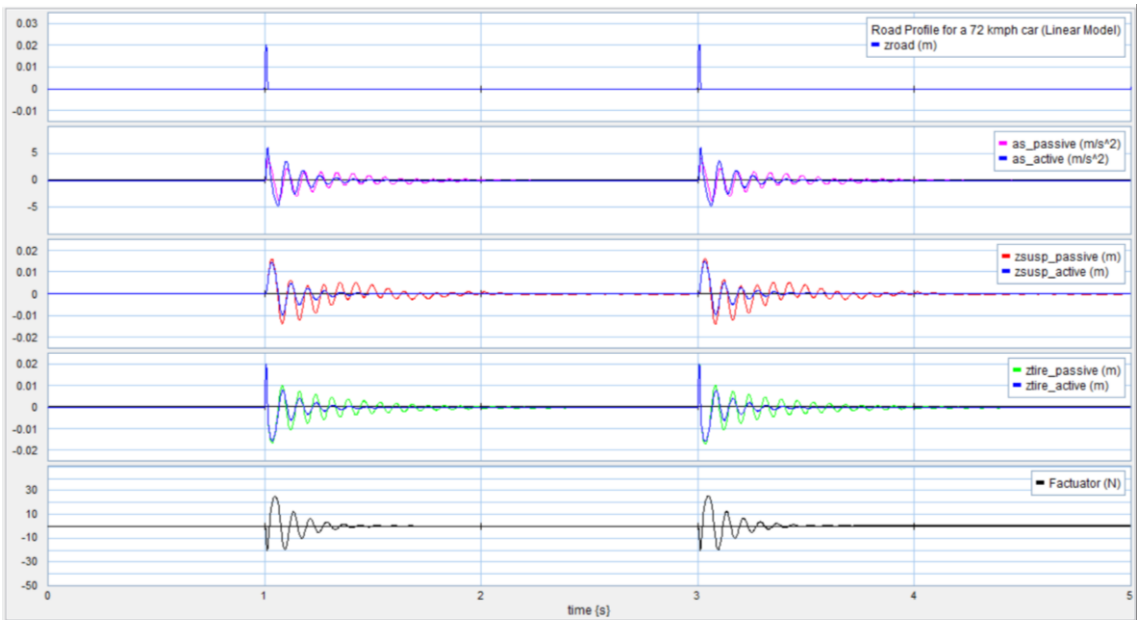


Figure 4.2: Controller performance for a 72 km/h car passing a 30 cm long 2 cm high road bump (linear model).

Using a feedback gain value corresponding to heavily weighted ride quality in the active model, as given in equation 4.17, and comparing the sprung mass accelerations from these two figures 4.2 and 4.3, it is observed that in the case of the linear model the sprung mass acceleration damps out after about 1 second whereas it becomes steady after about 0.7 seconds in the case of the nonlinear model, during the passive control. This happens probably due to the presence of Coulomb friction in the nonlinear model that makes the sprung mass steady more quickly. Hence, it is expected that the idealized controller would require less effort to attenuate the sprung mass acceleration for the nonlinear model. It is also evident from the actuator force plots found in these two figures. The peak force (25.4 N) required by the linear model is slightly higher than the peak force (25.2 N) required by the nonlinear model. The estimated peak force is achievable using the

available voice coil linear motor, described in the next chapter, during the experiment. The logarithmic decrement of the sprung mass acceleration indicates a damping ratio of 0.06 for the linear model and of 0.062 for the nonlinear model, during the active control. It also indicates that slightly more damping is present in the nonlinear system, which helps the idealized controller attenuate the sprung mass acceleration.

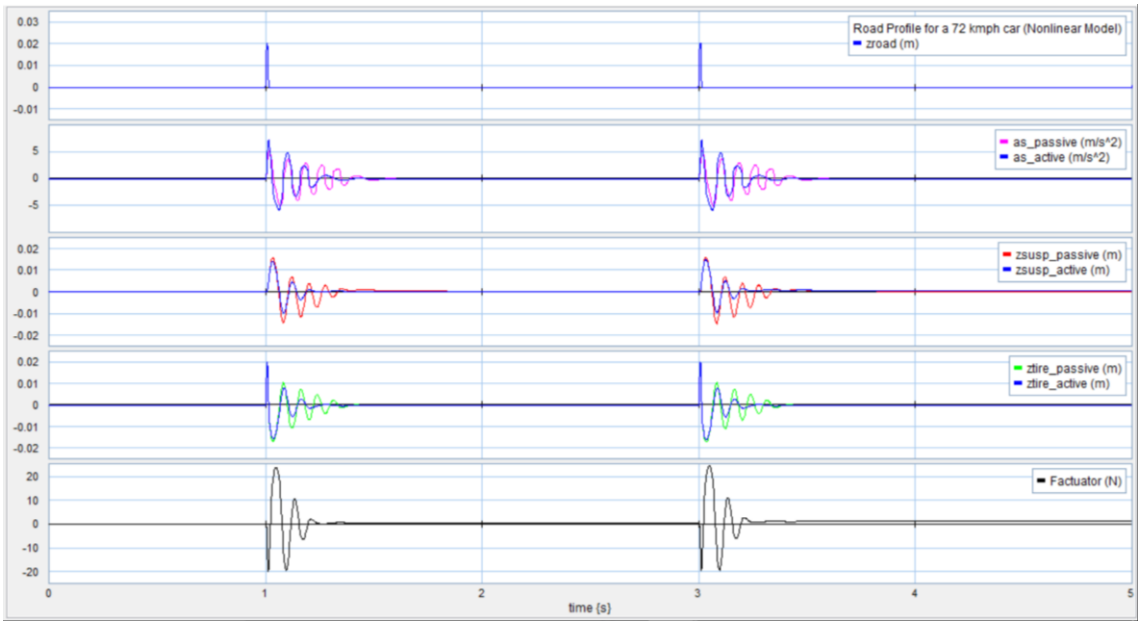


Figure 4.3: Controller performance for a 72 km/h car passing a 30 cm long 2 cm high road bump (nonlinear model).

The Bode plots for the linear and nonlinear models, generated at 1.15 seconds of the simulations, are shown in the figures 4.4 and 4.5 respectively. It is also observed from the Bode plots that for the linear model the sprung mass acceleration is slightly higher than that of the nonlinear model at sprung mass resonant frequency, during the passive control. The controller performs slightly better at low frequencies up to the sprung mass

resonance for the nonlinear model compared to that for the linear model. At higher frequencies, the controller performance is almost the same for both cases – linear and nonlinear.

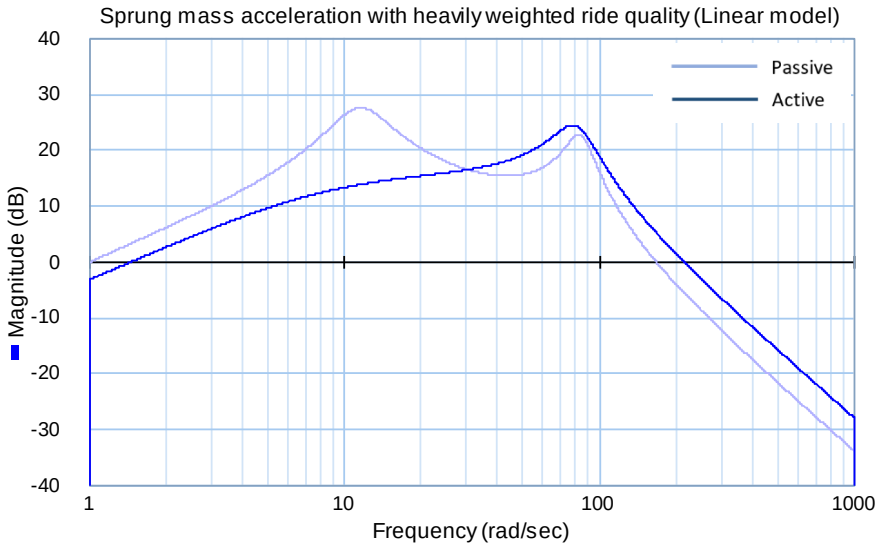


Figure 4.4: Controller performance for a 72 km/h car passing a 30 cm long 2 cm high road bump (nonlinear model).

The Bode plots related to the suspension deflection and the tire deflection are given in appendix D. The road holding performance of the controller is the same as discussed above for the sprung mass acceleration. Its performance related to the suspension deflection is almost unchanged for both cases – linear and nonlinear.

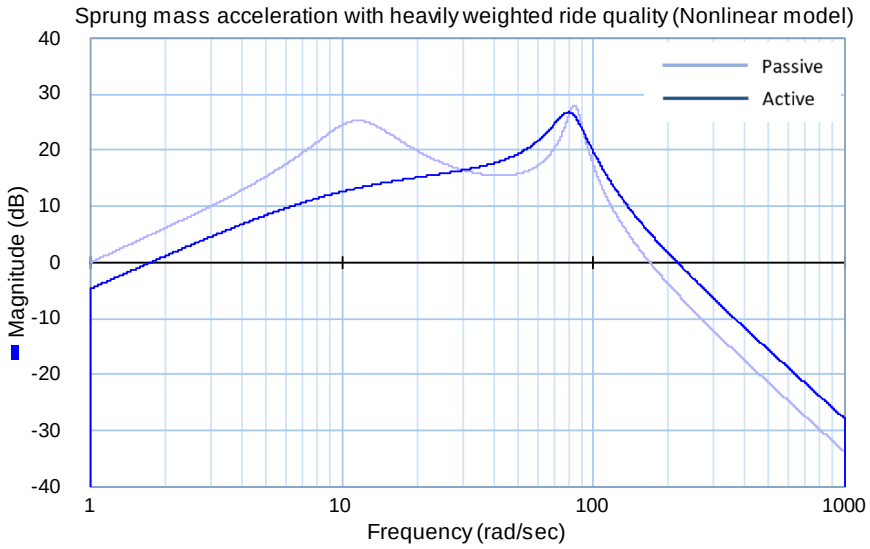


Figure 4.5: Controller performance for a 72 km/h car passing a 30 cm long 2 cm high road bump (nonlinear model).

4.4.2 Sinusoidal Road Inputs

Sinusoidal road inputs with high frequency low amplitude (2 mm 15 Hz and 2 mm 10 Hz) and low frequency high amplitude (10 mm 1 Hz and 10 mm 0.5 Hz) are used in simulation to study the performance of the idealized LQR controller for the nonlinear model.

Using a high frequency low amplitude (2 mm 15 Hz sine) road input, the passive nonlinear model generates two-peak Bode plots for all the three transfer functions of equations 3.19 – 3.21, as can be seen in figures 4.6 – 4.11. From the figures, the two peaks are observed at 11.48 rad/sec (sprung mass resonant frequency) and 84.38 rad/sec (unsprung mass resonant frequency). These two values are very close to previously calculated resonant frequencies (12.35 rad/s and 88.05 rad/s) in modal analysis of the test rig.

Using a feedback gain value corresponding to heavily weighted ride quality in the active nonlinear model, as given in equation 4.17, it is observed that the controller reduces the sprung mass acceleration considerably over a broad frequency range and eliminates the first resonance peak, as seen in figure 4.6. Its performance is almost same at unsprung mass resonance, compared to that of the passive suspension. Over a wide range of high frequencies, it even slightly increases the sprung mass acceleration, compared to that of the passive suspension.

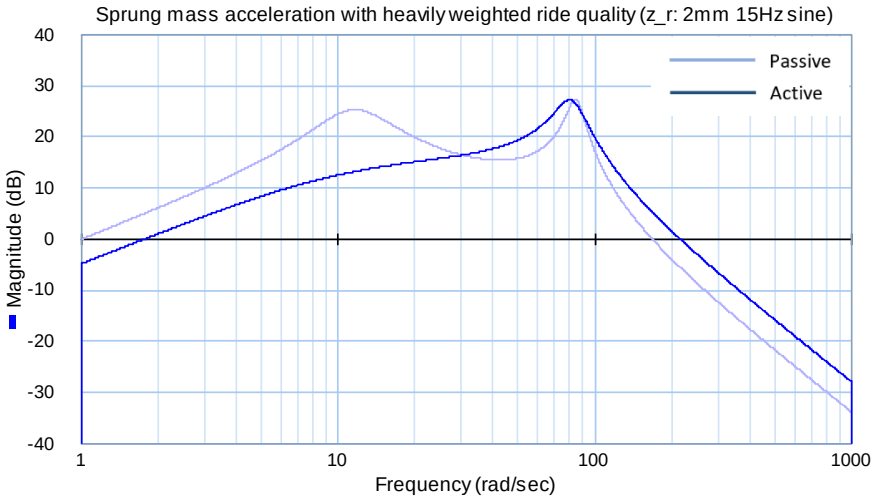


Figure 4.6: Sprung mass acceleration with heavily weighted ride quality for 2 mm 15 Hz (sine) road input (nonlinear model).

With heavily weighted ride quality, the suspension deflection performance is considerably worse than that of the passive suspension. It slightly reduces the suspension deflection at two resonant frequencies, but considerably increases the deflection at low frequencies, as shown in figure 4.7. However, it enhances the road holding performance, i.e., reduces

the tire deflection over a wide range of frequencies including two resonant frequencies, as shown in figure 4.8.

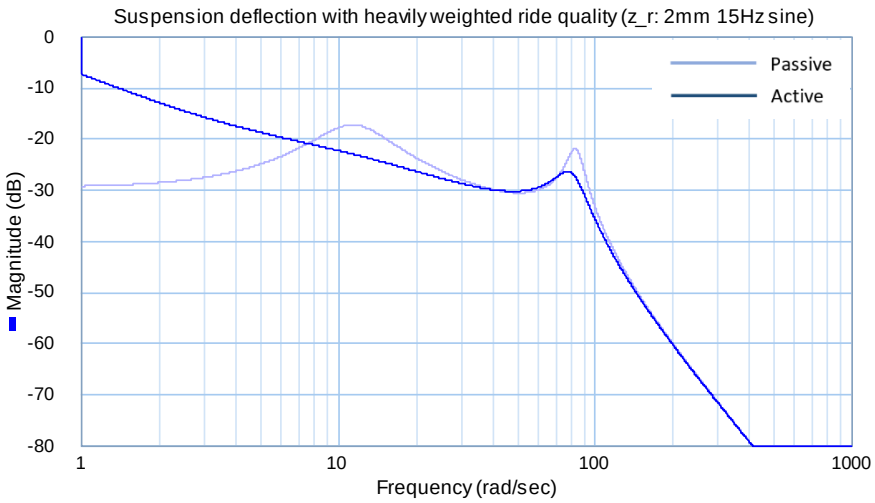


Figure 4.7: Suspension deflection with heavily weighted ride quality for 2 mm 15 Hz (sine) road input (nonlinear model).

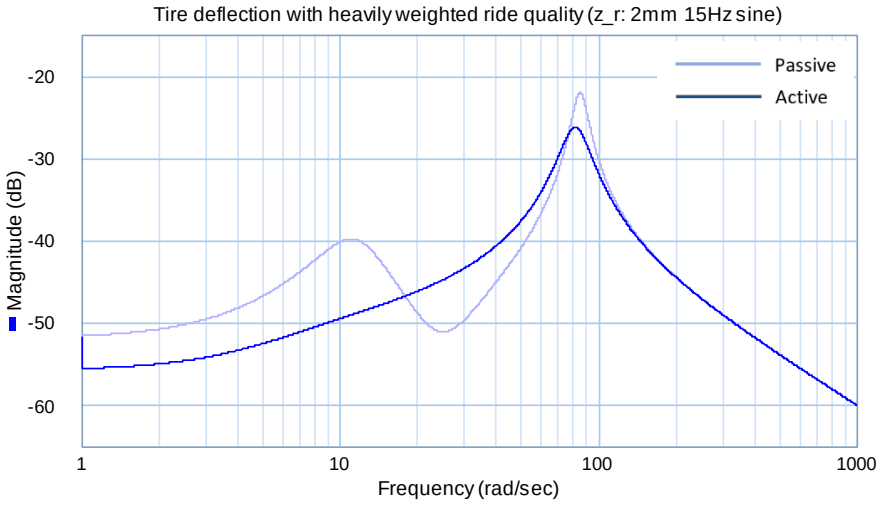


Figure 4.8: Tire deflection with heavily weighted ride quality for 2 mm 15 Hz (sine) road input (nonlinear model).

Using the feedback gain value from equation 4.18 in the active nonlinear model for moderately weighted ride quality, the sprung mass acceleration decreases at first resonance, but performance is almost same at unsprung mass resonance. However, a new resonance of lower magnitude is observed at 4.45 rad/sec during the active control, as can be seen in figure 4.9.

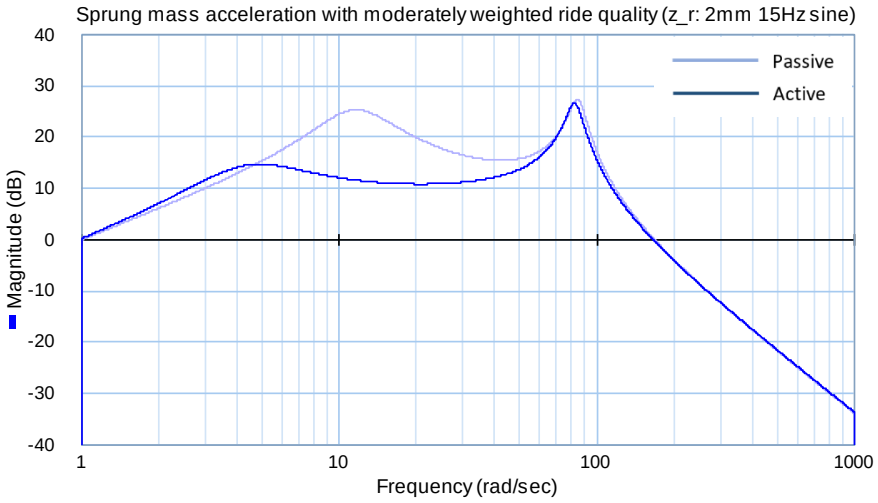


Figure 4.9: Sprung mass acceleration with moderately weighted ride quality for 2 mm 15 Hz (sine) road input (nonlinear model).

For moderately weighted ride quality, the suspension deflection performance is even worse. Though it slightly reduces the deflection at sprung mass resonant frequency, but it introduces a resonance of higher magnitude at 4.45 rad/sec, as shown in figure 4.10. It improves the tire deflection at sprung mass resonance, but performance is almost same at unsprung mass resonance. A resonance of very low magnitude is observed at 4.45

rad/sec, as shown in figure 4.11. The plots of a similar analysis with 2 mm 10 Hz (sine) road input are given in appendix D, showing the similar results.

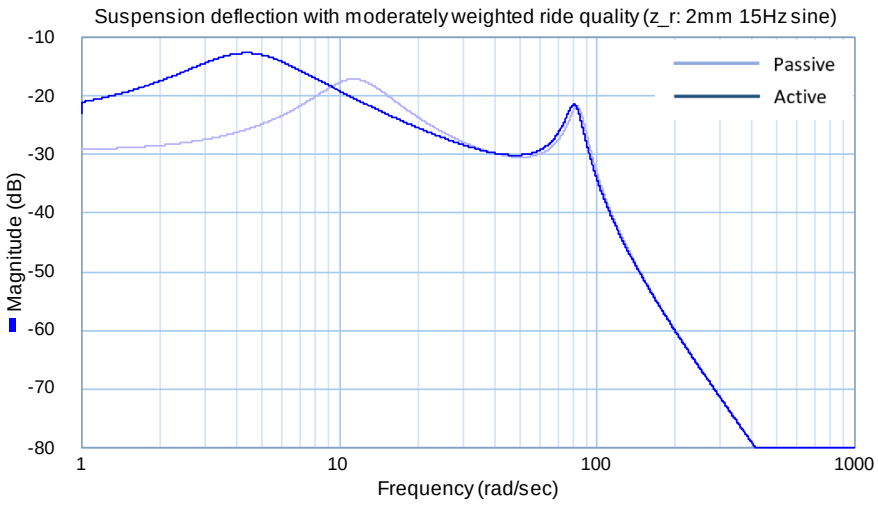


Figure 4.10: Suspension deflection with moderately weighted ride quality for 2 mm 15 Hz (sine) road input (nonlinear model).

Again, the similar simulation is done for some high amplitude low frequency road inputs of 10 mm 0.5 Hz (sine) and 10 mm 1 Hz (sine). Surprisingly for both models- nonlinear passive and active, one peak resonance curve is found, as shown in figures 4.12 – 4.17. The resonance occurs at a unique frequency of about 41 rad/s which is not equal to either of the system natural frequencies.

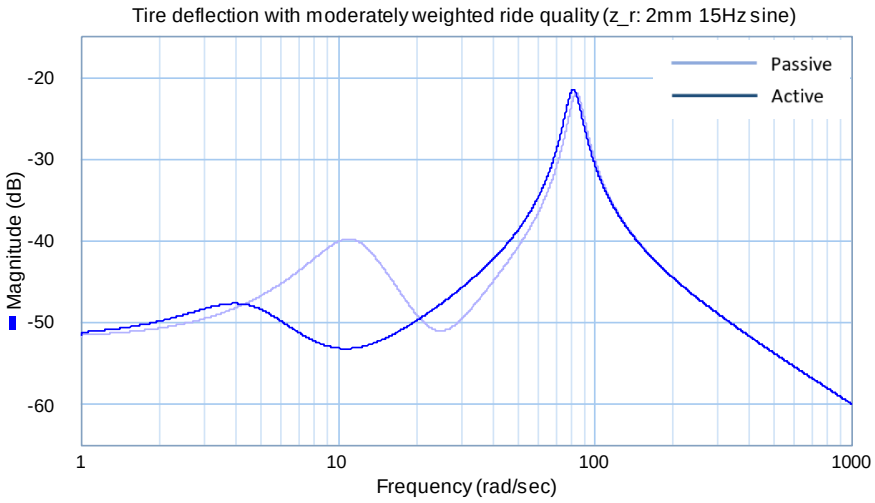


Figure 4.11: Tire deflection with moderately weighted ride quality for 2 mm 15 Hz (sine) road input (nonlinear model).

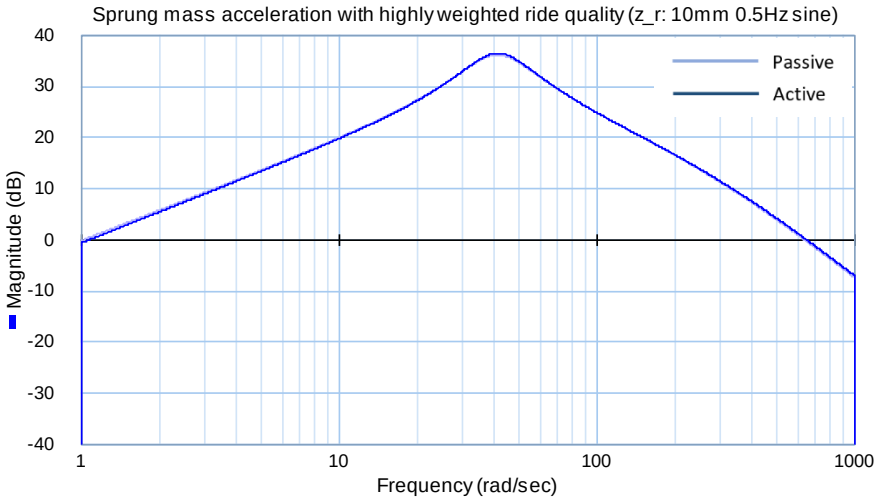


Figure 4.12: Sprung mass acceleration with heavily weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model).

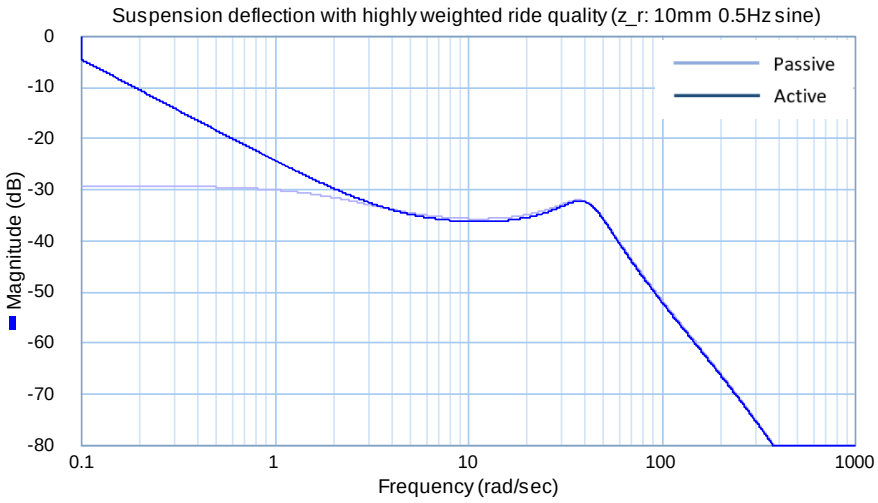


Figure 4.13: Suspension deflection with heavily weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model).

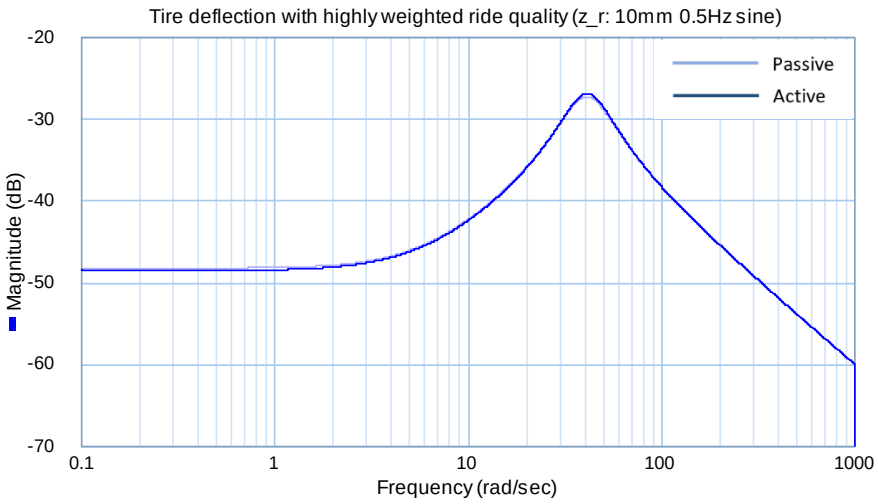


Figure 4.14: Tire deflection with heavily weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model).

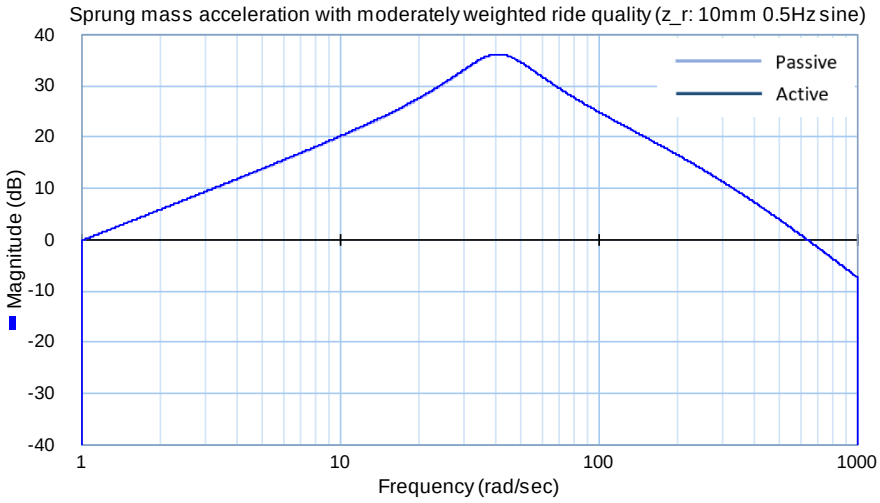


Figure 4.15: Sprung mass acceleration with moderately weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model).

In figures 4.12 – 4.17, the two plots for the active and passive models are too close and hardly separable. The idealized controller (based on linear assumptions) has almost no effect on ride quality, suspension deflection and tire deflection of the nonlinear model for these low frequency road inputs (except only it increases the suspension deflection at low frequencies), compared to that of the passive suspension, as can be seen in figures 4.12 – 4.17. The plots for the nonlinear model with 10 mm 1 Hz (sine) road input are given in appendix D, showing the similar results.

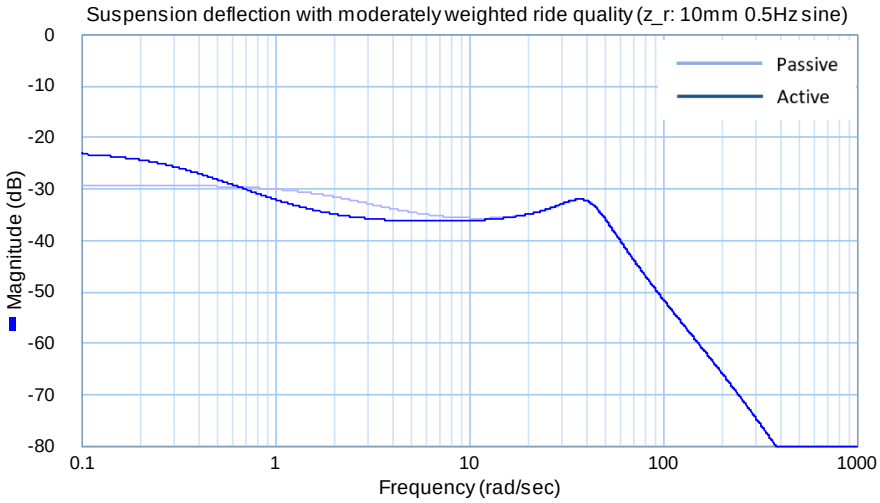


Figure 4.16: Suspension deflection with moderately weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model).

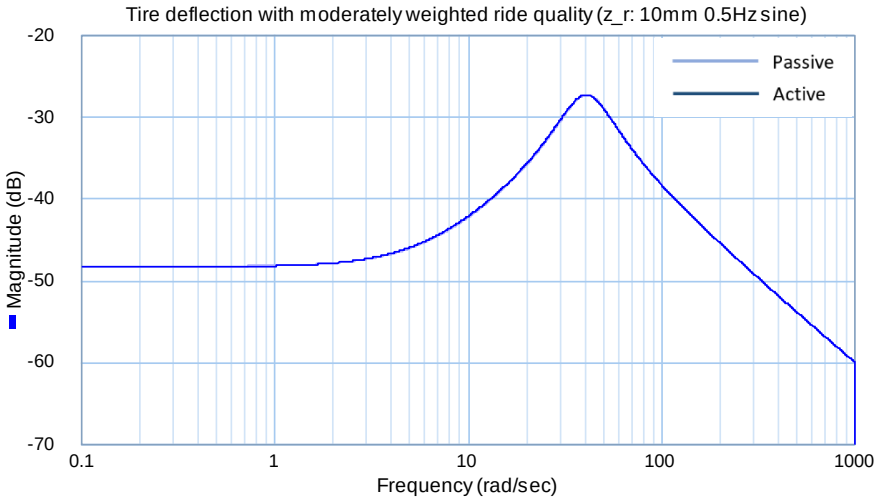


Figure 4.17: Tire deflection with moderately weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model).

One of the reasons behind these one-peak Bode plots can be the presence of nonlinearity in the shock absorber. To verify this, the following 5 cases are investigated using a 10 mm 0.5 Hz sine road input for sprung mass acceleration-

- Case 1: linear suspension (25 Ns/m), linear rail friction (10 Ns/m) – linear model.
- Case 2: linear suspension (25 Ns/m), nonlinear rail friction.
- Case 3: nonlinear suspension, nonlinear rail friction – nonlinear model.
- Case 4: nonlinear suspension, linear rail friction (10 Ns/m).
- Case 5: nonlinear suspension, no rail friction.

The corresponding plots can be found in appendix D. From the plots, it can be seen that only the first two cases with the linear suspension show two-peak Bode plots for sprung mass acceleration. The presence of nonlinear or linear rail friction, or the absence of rail friction has no effect on the system natural frequencies. Hence, the nonlinearity present in the suspension is the possible reason behind these one-peak Bode plots for low frequency road inputs.

Since the nonlinear suspension possibly causes these one-peak Bode plots, to show this, the simulation is again done using a suspension damper with linear damping coefficient of 25 Ns/m for these low frequency road input cases, while the nonlinear rail frictions are still present. This time for the same controller, the expected two-peak curves are obtained for the passive model, as can be seen in figures 4.18 – 4.23. For both heavily and moderately weighted ride quality, the active model reduces the sprung mass acceleration

over a wide range of frequencies compared to that of the passive model, as shown in figures 4.18 and 4.21. It even increases the sprung mass acceleration at high frequencies for heavily weighted ride quality and introduces a new resonance of lower magnitude at 4.45 rad/sec for moderately weighted ride quality. Comparing the figures 4.6 and 4.18, it can be said that the sprung mass is slightly less excited during the passive control at unsprung mass resonant frequency for low frequency road input.

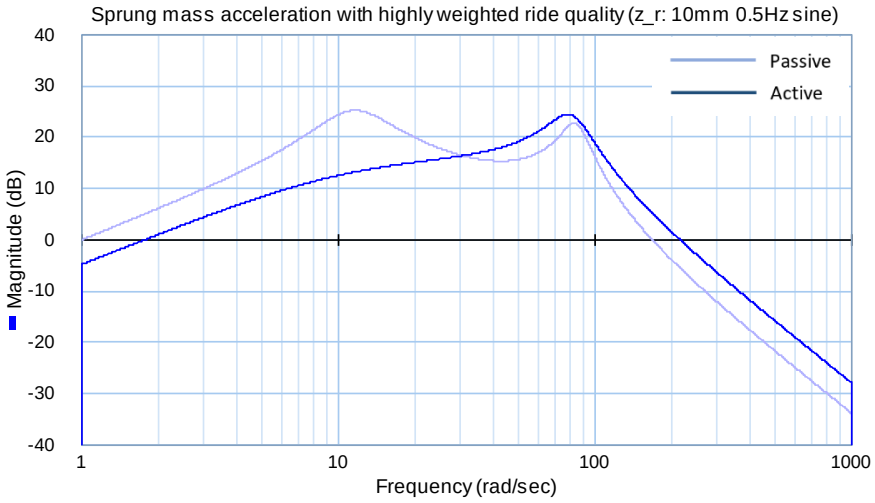


Figure 4.18: Sprung mass acceleration with heavily weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model with a linear suspension damping = 25 Ns/m).

For both cases of ride quality, it shows slight improvement in the suspension deflection at two resonant frequencies, but decreases the performance at low frequencies, as shown in figures 4.19 and 4.22. The active control exhibits a resonance of higher magnitude, than that of the sprung mass resonance during the passive control, at 4.45 rad/sec for moderately weighted ride quality (figure 4.22).

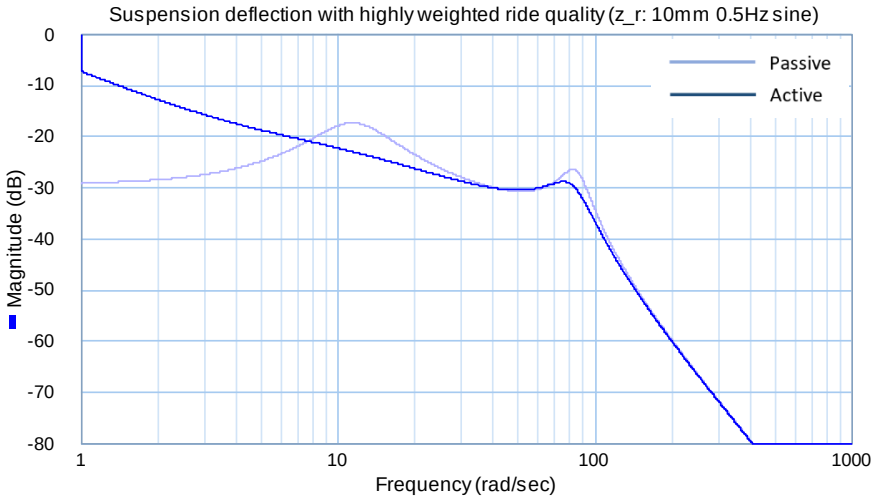


Figure 4.19: Suspension deflection with heavily weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model with a linear suspension damping = 25 Ns/m).

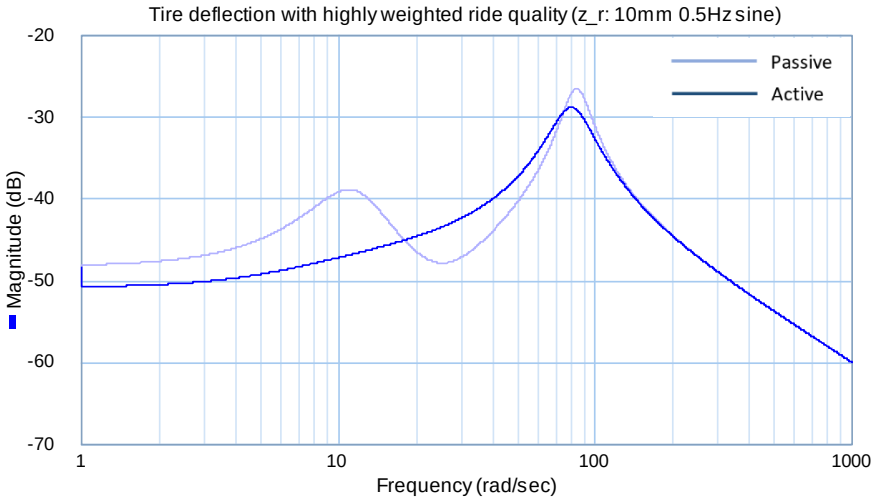


Figure 4.20: Tire deflection with heavily weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model with a linear suspension damping = 25 Ns/m).

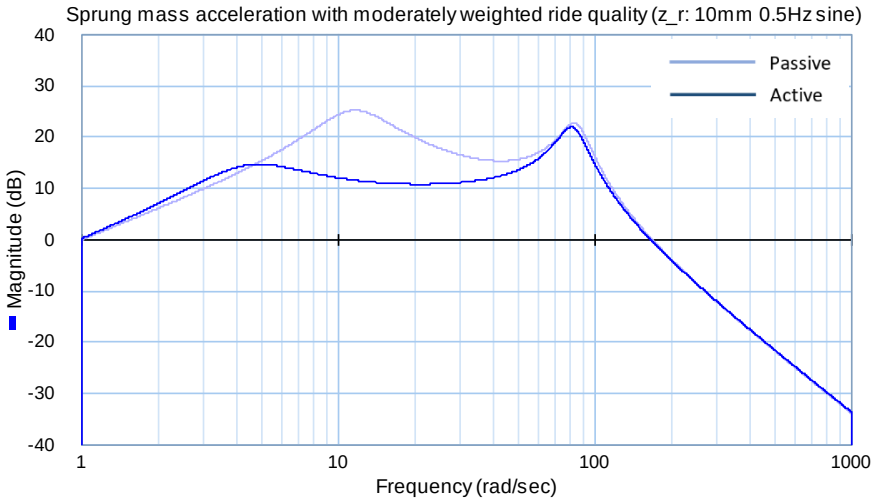


Figure 4.21: Sprung mass acceleration with moderately weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model with a linear suspension damping = 25 Ns/m).

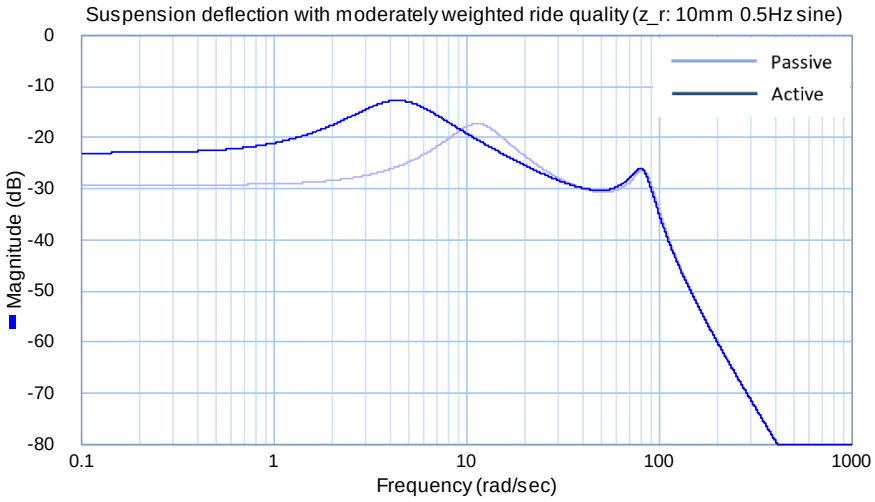


Figure 4.22: Suspension deflection with moderately weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model with a linear suspension damping = 25 Ns/m).

For both cases of ride quality, overall it shows better road holding performance compared to that of the passive suspension, as shown in figures 4.20 and 4.23. For moderately

weighted ride quality, the active control introduces a resonance of lower magnitude, than that of the sprung mass resonance during the passive control, at 4.45 rad/sec (figure 4.23).

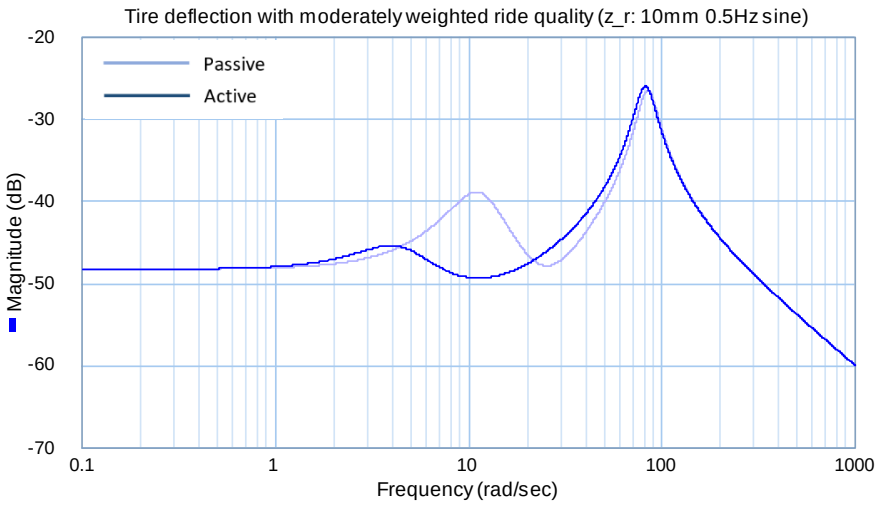


Figure 4.23: Tire deflection with moderately weighted ride quality for 10 mm 0.5 Hz (sine) road input (nonlinear model with a linear suspension damping = 25 Ns/m).

According to the above analysis, it can be said that for low frequency road inputs the nonlinear damper introduces a unique resonant frequency (41 rad/sec) in the system and degrades the controller performance.

The actuator force plots for all cases described above are given in appendix D. The estimated maximum force is found to be -11.8 N, which would be achievable using the available voice coil linear motor if the system were to be implemented in hardware, described in the next chapter, during the experiment.

The voice coil actuator will be introduced in the test setup as an attempt to make the system active. The next chapter will discuss the experiments related to the verification and assessment of its static and dynamic responses.

Chapter 5: Developing the Interface between Simulation and Hardware

A linear voice coil motor will be used in the test rig to make the suspension model active. However, before introducing the motor in the quarter car setup and the controller in the control model, it is important to verify the responses of the motor in different static and dynamic conditions. Simple open loop tests are done just to verify the force constant of the actuator. Then closed loop control tests are done to verify its dynamic responses in real-time using position and force control signals.

This chapter will first describe the instrumentation and software required for these tests, and then it will explain the experimentation and the test results. The hardware to interface the simulation software with the quarter car physical test rig, and to allow control design in 20sim, had not been used before in the research group, and its initial use and verification is a contribution of this thesis.

The hardware and software used in these experiments are-

- Hardware
    - Voice coil actuator,
    - Load cell,
    - Instrumentation amplifier,
    - Linear potentiometric displacement transducer (LPDT),
    - Motor driver,
    - Data acquisition board,
    - Computer board.
- Software
    - 20-sim,
    - 20-sim 4C

Figure 5.1 shows the schematic of software-hardware interface. The 20-sim model is exported to the 20-sim 4C, which compiles the model as C-code into the computer board. The computer board provides necessary PWM signal to the motor driver. The motor driver drives the actuator and controls its direction. The sensor responses are acquired and sent to the computer board by the data acquisition board.

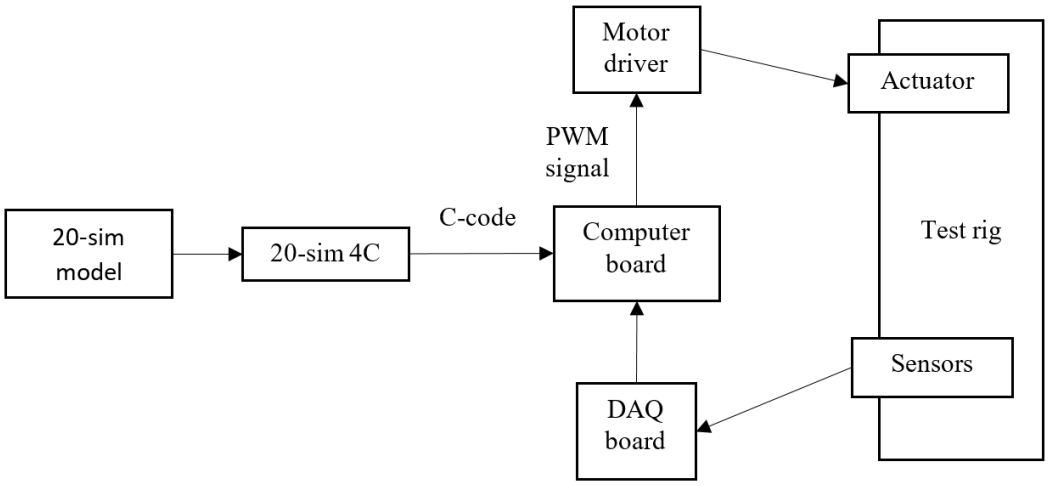


Figure 5.1: Schematic of software-hardware interface.

5.1 Hardware

5.1.1 Voice Coil Actuator

A linear voice coil motor (manufacturer: Moticont, model: LVCM-051-089-01) is used as an actuator in the test rig. It has a diameter of 50.8 mm and a length of 130.6 mm. The stroke length of the motor is 57.2 mm. The motor requires a maximum continuous power of 40 W. The coil resistance and inductance are 6 Ohms and 2.7mH (at 1000 Hz) respectively. It generates a continuous force of 26.2 N and an intermittent force of 82.7 N at 10% duty cycle with a force constant of 10.1 N/A. The masses of the body and coil assembly are 1155 grams and 195 grams respectively. All values are specified at 25 °C [41].

Figure 5.2: Linear voice coil motor (model: LVCM-051-089-01) [41].

5.1.2 Load Cell

A “S” beam load cell (manufacturer: OMEGA Engineering Inc., model: LCCA-250) is used to measure the required force during the tests. It is made of nickel plated carbon steel and it has a rated capacity of 250 lbs. The excitation voltage required is 10 Vdc (15 Vdc maximum). It gives an output at 3 mV/V sensitivity. The accuracy is 0.037% of full scale and the bias is 1% of full scale. The operating temperature should be in between 0 to 150 °F [42].

Figure 5.3: “S” beam load cell (series: LCCA) [42].

5.1.3 Instrumentation Amplifier

The load cell output is amplified using an instrumentation amplifier (manufacturer: Analog Devices Inc., model: AD620). Its operating voltage required is ±18 V maximum. It has a very low internal power dissipation of 650 mW and a very low noise of 9 nV/√Hz at 1 kHz. It can support a differential input voltage of ±25 V maximum. The operating temperature should be in between -40 to 85 °C [43].

The gain resistor (RG) for a gain (G) can be calculated using the following formula,

$$R_G = \frac{49.4\ k\Omega}{G_1 - 1} \tag{5.1}$$

Two resistors of 109.6 Ohm and 90.7 Ohm are used in parallel as an equivalent gain resistor (RG) of 49.6 Ohm. For this gain resistor value, the gain (G1) calculated using the above equation is 996.97. Based on this gain value, the calibration equation found for the load cell, using several known weights, is,

$$\begin{aligned} F\ (N) &= 36.7 * V\ (V) + 12.46;\ [vertical\ position] \\ F\ (N) &= 36.7 * V\ (V) + 11.17;\ [horizontal\ position] \end{aligned} \tag{5.2}$$

These calibration equations give a positive value of force for tension and a negative value of force for compression.

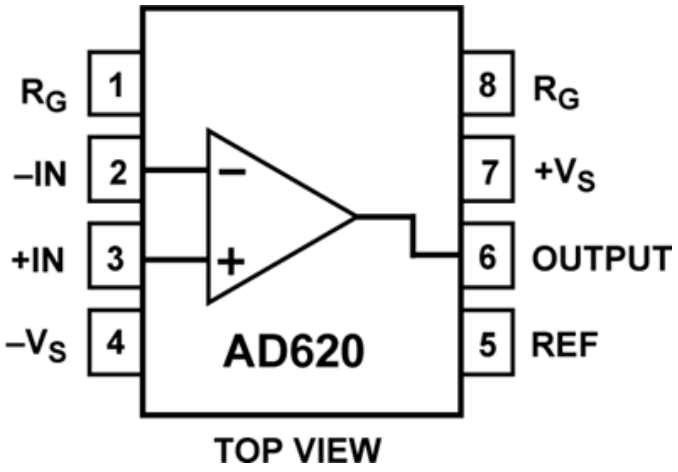


Figure 5.4: Instrumentation amplifier (model: AD620) [43].

5.1.4 Linear Potentiometric Displacement Transducer (LPDT)

Two linear potentiometric displacement transducers (manufacturer: MEGATRON Elektronik GmbH & Co., model: RC13-75M) are used to measure the suspension deflection and the tire deflection during the experiments. It has a robust aluminium housing with 13 mm diameter and comes with two mounting brackets. The guided push rod has an electrical travel of 75 mm and a mechanical travel of 80 mm, and moves at a speed up to 10 m/s. The operating temperature should be in between -30 to 100 °C [44].

Based on its voltage outputs at its two terminal points (0 mm and 80 mm), a calibration equation for the LPDT is developed as follows,

$$L\ (m) = (7.93 * V\ (V) - 0.02)\ /\ 1000 \qquad (5.3)$$

Figure 5.5: Linear potentiometric displacement transducer (series: RC13) [44].

5.1.5 Motor Driver

A DC motor driver (manufacturer: Cytron Technologies, model: MD10C Rev3.0) is used to drive the actuator using the PWM (Pulse Width Modulation) output signal from the controller board. It can take 3.3 V and 5 V logic level input, and supports motor voltage starting from 5 to 30 V with a maximum current up to 13 A continuous and 30 A peak (for 10 seconds). It keeps the output PWM frequency the same as the input PWM frequency and supports PWM frequencies up to 20 kHz. The output PWM signal has a voltage amplitude equal to the excitation voltage. Sign-magnitude PWM mode is used to drive the motor. In this mode, it takes two control signals to control the speed and the direction of the motor. PWM pin takes the PWM signal that controls the speed of the motor while the DIR pin binary input controls the direction of the motor. Two LED (A and B) indicate the direction of the motor [45].

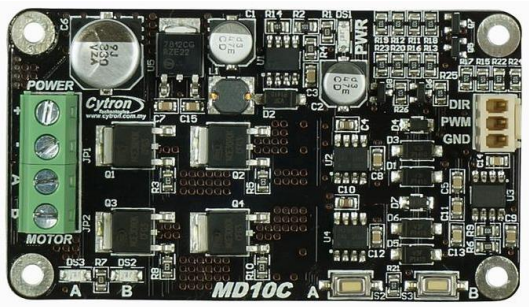


Figure 5.6: DC motor driver (series: MD10C) [45].

5.1.6 Data Acquisition Board

A data acquisition board (manufacturer: Technologic Systems, model: TS-ADC16) is used to collect the sensor data. It has 16 channels of 16-bit analog to digital conversion, 4 channels of 12-bit digital to analog conversion, 4 digital inputs and 1 digital output, and selectable voltage ranges for the 16 ADC channels up to ±10 V in single or differential mode. One digital pin outputs 0 and 3.3 V. It takes two ADC samples simultaneously at a sampling rate up to 2 x 100 ksps [46]. This board is connected to a computer board.

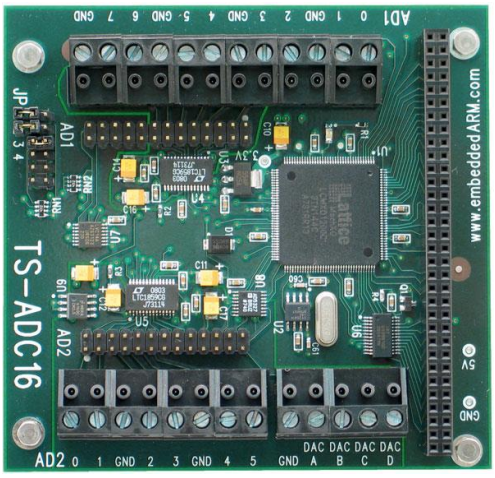


Figure 5.7: Data acquisition board (model: TS-ADC16) [46].

5.1.7 Computer Board

A single board computer (manufacturer: Technologic Systems, model: TS-7300) is used to process the sensor data and provide the calculated output to the system. It has a 200 MHz ARM9 processor from Cirrus with MMU (Memory Management Unit) and 32 MB SDRAM. The operating temperature should be in between -40 to 85 °C. The board has 5 channels of 12-bit analog to digital conversion with an input range of 0 to 3.3 V. It has 55 total DIO (Digital Input/Output) lines available on three headers labeled 'DIO1', 'DIO2' and 'LCD'. The DIO1 and DIO2 headers have 3.3 V power available while the LCD header has 5 V power. The 20-pin DIO2 header has 18 DIO lines implementing two XDIO ports (XDIO1 and XDIO2), as shown in figure 5.9, which are controlled by the on-board FPGA (Field-Programmable Gate Array) [47]. These XDIO ports can provide the necessary PWM and binary direction signals to the motor driver.

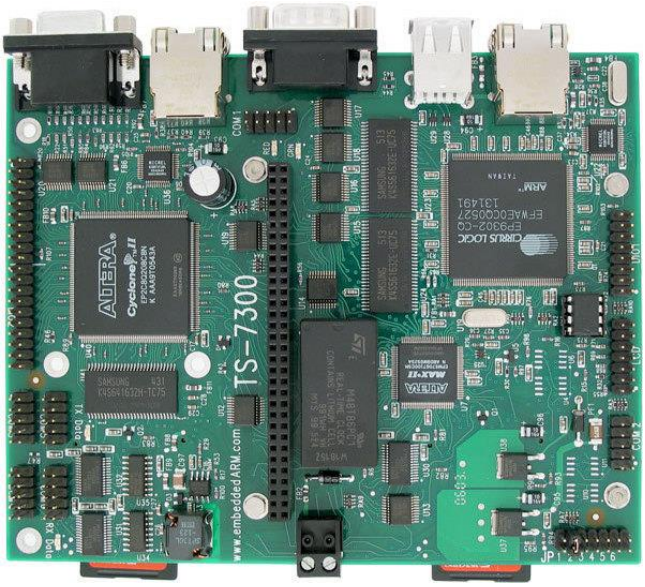


Figure 5.8: Single board computer (model: TS-7300) [47].

	STEPPER2_nDIR PWM2_nDIR	STEPPER2_pDIR PWM2_pDIR	STEPPER2_PULSE PWM2_DUTY	EDGE2_I QUAD2_I	QUAD2_B	EDGE2_CNTR QUAD2_A			
GND	XDIO2_2	XDIO2_3	XDIO2_4	XDIO2_5	XDIO2_6	XDIO2_7	RX_LED	TX_LED	+3.3V
22	24	26	28	30	32	34	36	38	40
21	23	25	27	29	31	33	35	37	39
XDIO1_0	XDIO1_1	XDIO1_3	XDIO1_3	XDIO1_4	XDIO1_5	XDIO1_6	XDIO1_7	XDIO2_0	XDIO2_1
		PWM1_nDIR STEPPER1_nDIR	PWM1_pDIR STEPPER1_pDIR	PWM1_DUTY STEPPER1_PULSE	QUAD1_I EDGE1_I	QUAD1_B	QUAD1_A EDGE1_CNTR	CAN1	CAN2

Figure 5.9: TS-7300 DIO-2 pin layout [48].

5.2 Software

5.2.1 20-sim

20-sim is a bond graph method based modeling and simulation software. It helps simulate and analyze the behaviour of multi-domain dynamic systems, i.e., mechanical, electrical and hydraulic systems etc. or any combination of these, and create control systems. 20-sim has a vast library of system components. Using these components and the knowledge of the bond graph, it allows user to develop the system model graphically, and quickly and intuitively without compromising the simulation power [8].

In this research, 20-sim is used to develop simulation and control models for different test rig experiments.

5.2.2 20-sim 4C

20-sim 4C provides a rapid prototyping environment for the control models. It runs hand-written or automatically generated C-code on hardware like embedded ARM based processor boards, PC/104 systems etc. It can import models as C-code from 20-sim, Simulink and Scilab. Hence, it allows to operate and read sensors, control dynamic

systems in real-time. The system parameters can be modified during run-time in 20-sim 4C [49].

In this research, 20-sim 4C is used to import 20-sim models as C-code, export that C-code to the target TS-7300 board and operate the controller.

5.3 Actuator Test

The actuator has a force constant of 10.1 N/A. A simple open loop test is done just to verify its force constant in real applications. Then PID based closed loop control tests are conducted to control the movement of the actuator and the force generated by the actuator using LPDT and load cell data respectively. Both open loop and closed loop tests are done in the horizontal position of the test rig. The test setup used for both types of test is shown in figure 5.10.

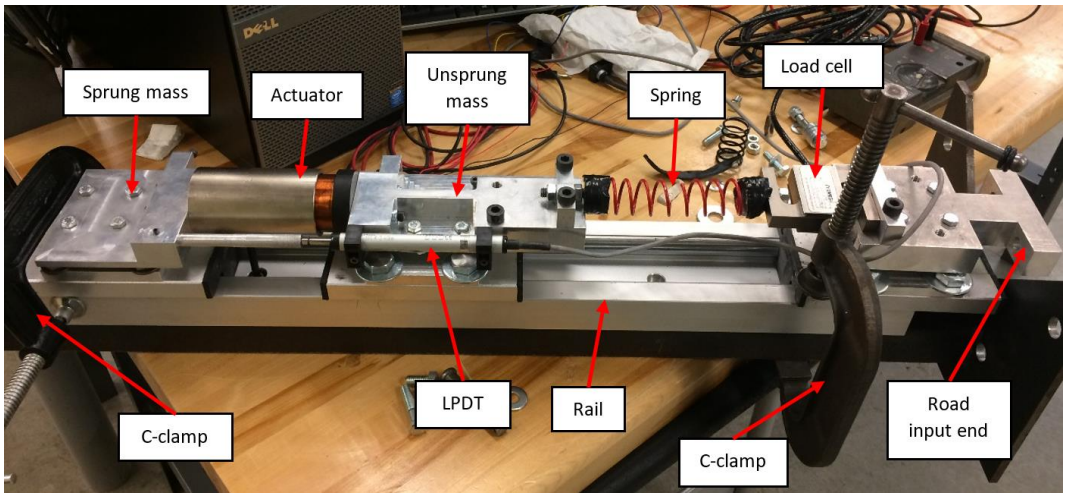


Figure 5.10: Test setup for open loop and closed loop control tests of the actuator.

As can be seen in figure 5.10, the sprung mass and the road input end of the test rig are kept fixed using two C-clamps while the unsprung mass is the only moving mass on the rail. The actuator is placed in between the sprung and unsprung masses, and it exerts force on the unsprung mass. A spring is connected to the other end of the unsprung mass. The load cell is placed in between the spring and the road input end of the test rig. The LPDT measures the displacement of the actuator, i.e., the deflection of the spring while the load cell measures the spring force. Using equation 4.8, the system natural frequency is found to be approximately 4.1 Hz, as the spring constant of the spring used is 1663.1 N/m and the moving mass (the unsprung mass and the actuator coil mass) attached to the spring is 2.5 kg.

5.3.1 Open Loop Test

The responses of the actuator are studied in response to an open loop sinusoidal voltage input of 14 V with several frequencies (0.1, 0.5, 1.0 and 5.0 Hz) at 3 different positions of its stroke length. The peak current observed is about 2.14 A. Since the force constant of the actuator is 10.1 N/A, the peak force of the actuator is expected to be about 21.6 N and the peak expected deflection of the spring (k = 1663.1 N/m) corresponding to that force is about 13 mm (which is calculated neglecting the frictional loss due to the unsprung mass motion on the rail).

First, the actuator is set at about its mid-stroke (L = 30 mm). Now, it is given an excitation voltage of 14 V 0.1 Hz sine, i.e., the frequency of the actuator force is 0.1 Hz which is lower

than the system natural frequency of 4.1 Hz. Hence, it is expected that the unsprung mass motion would be in phase with the force, i.e., the voltage or the current. It can be seen in figure 5.11 that the LPDT displacement, i.e., the unsprung mass motion is almost in phase with the current. Similar tests are done using voltage input of 14 V 0.5 Hz and 1.0 Hz, and also at two other positions of the actuator stroke (L = 14 and 45 mm). The corresponding plots are included in the appendix E. Again, it is observed that the unsprung mass motion is almost in phase with the current. The negligible amount (about 0.28 seconds) of phase shift happens probably due to the delay in the computation.

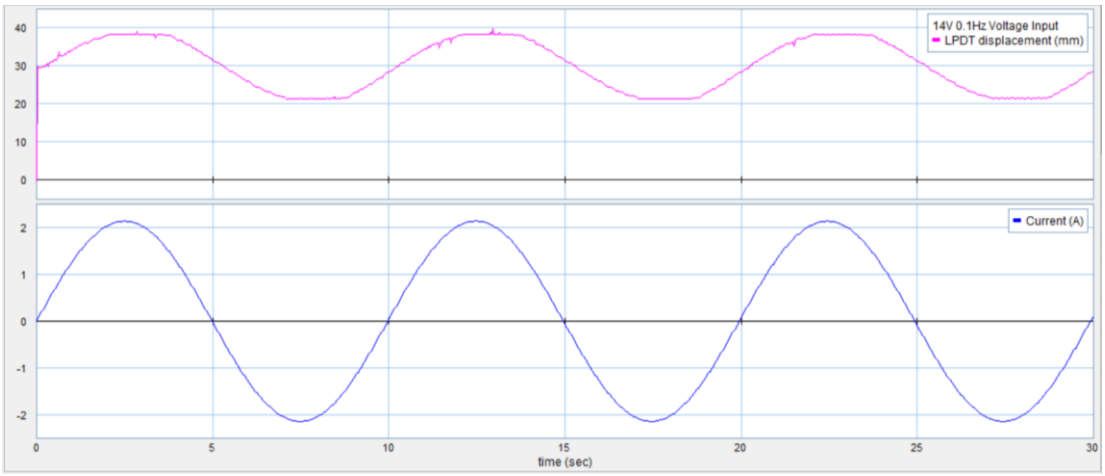


Figure 5.11: Open loop actuator test using 14 V 0.1 Hz sine input voltage.

In all the cases, the peak spring deflection is found to be in between 8.1 to 8.7 mm, from figure 5.11 and appendix E, which is supposed to be about 13 mm. But a smaller value is expected in this case, since the inertial effect of the unsprung mass is less due to the low frequency of the force and hence the effect of the Coulomb friction present in between

the unsprung mass and the rail is high. However, it can be said that the actuator works the same throughout its stroke length.

Again, a voltage input of 14 V 5.0 Hz sine is given to the actuator. Since the excitation force frequency of 5.0 Hz is higher than the system natural frequency of 4.1 Hz, it is expected that the unsprung mass motion would be 180 degrees out of phase with the force, i.e., the current and it is also evident from the figure 5.12.

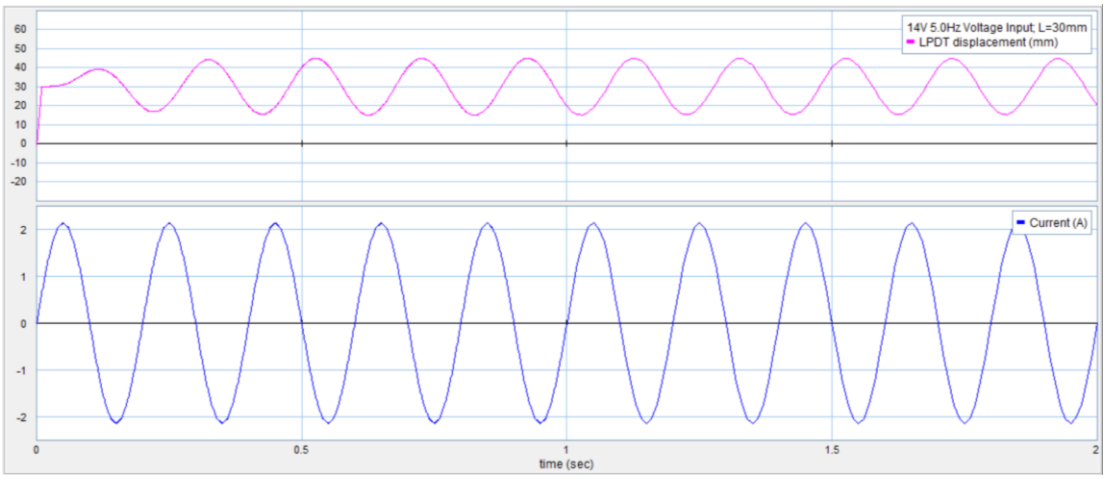


Figure 5.12: Open loop actuator test using 14 V 5.0 Hz sine input voltage.

In this case, the peak spring deflection is found to be about 14.8 mm, from the figure 5.12, which is expected because the inertial effect of the unsprung mass is high as the frequency of the force is high and hence the effect of the Coulomb friction is less.

Now, it can be said that the designer specified value of 10.1 N/A for the motor is safe to use as the force constant of the motor in the control model.

5.3.2 Closed Loop Position Control

The test setup, shown in figure 5.10, is modeled using 20-sim, as shown in figure 5.13, to test the response of the actuator in response to a reference displacement signal (step and sine input). The model includes a sub-model (ControlSystem) as the control system which provides a means of doing a closed loop position control of the actuator using a PID based controller, as shown in figure 5.14.

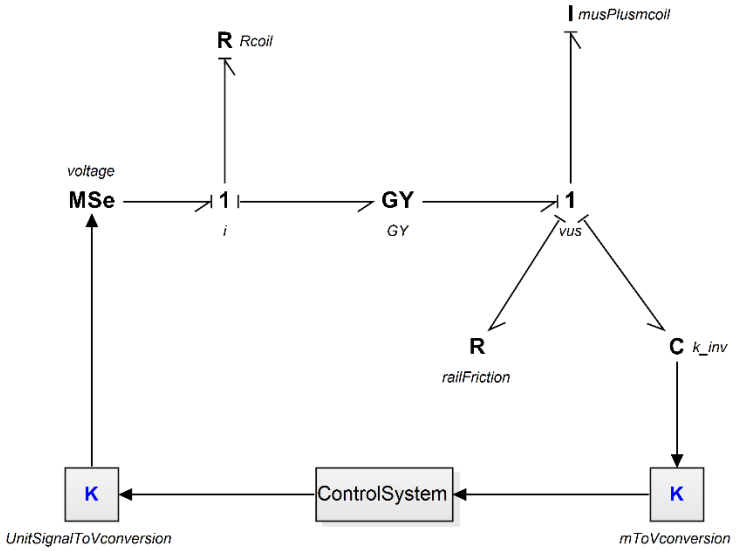


Figure 5.13: Position control model for actuator test with sub-model 'ControlSystem'.

The model helps simulate and analyze the responses of the actuator for different reference signals. Only the sub-model 'ControlSystem' is exported to 20-sim 4C as C-code and it helps analyze the actuator responses in real-time experiments for the same reference signals.

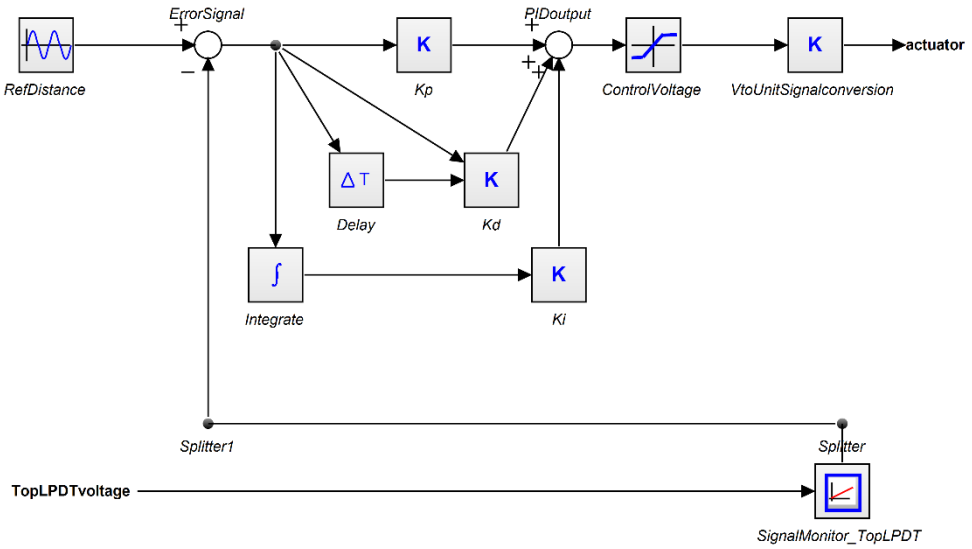


Figure 5.14: Actuator position control sub-model (ControlSystem).

As shown in the figures 5.13 and 5.14, the sub-model 'ControlSystem' takes in the spring (k_inv) state - displacement, i.e., the LPDT data as 'TopLPDTvoltage', in terms of voltage calculated using the LPDT calibration equation 5.3 and converted inside the 'mToVconversion' block. Inside the sub-model, this voltage (TopLPDTvoltage) is again changed back to the displacement unit inside the 'SignalMonitor_TopLPDT' block using the same equation and subtracted from the reference displacement signal (RefDistance) to calculate the error signal. The error signal is then passed through the PID blocks (Kp, Ki, Kd). The PID output is kept limited to generate a safe control voltage inside the 'ControlVoltage' block. This voltage is then converted to a PWM signal by the TS-7300 board and passed to the actuator through the motor driver during the experiment. During the simulation, the control voltage signal is passed to the 'MSe' element in the model. The

20-sim codes for the model elements are included in the appendix B. In this model, the standard 20-sim PID block is not used as it introduces dependent states, and 20-sim 4C requires an explicit model. The use of time delay block (Delay) and numerical differentiation solves that problem, as shown in the figure 5.14.

First, a step reference displacement signal is used. The signal stays at zero for first one second. Then for next 5 seconds, it indicates a 3-mm extension of the spring followed by a 13-mm compression of the spring, as can be seen in figure 5.15. Simulation shows that, with a PID gain of 500, 10000, 30 respectively, the response of the model can approximately follow the reference signal. From the figure 5.15, the response looks similar to a second-order overdamped system with a rise time of about 0.25 seconds. The maximum control voltage is expected to be about 10 V.

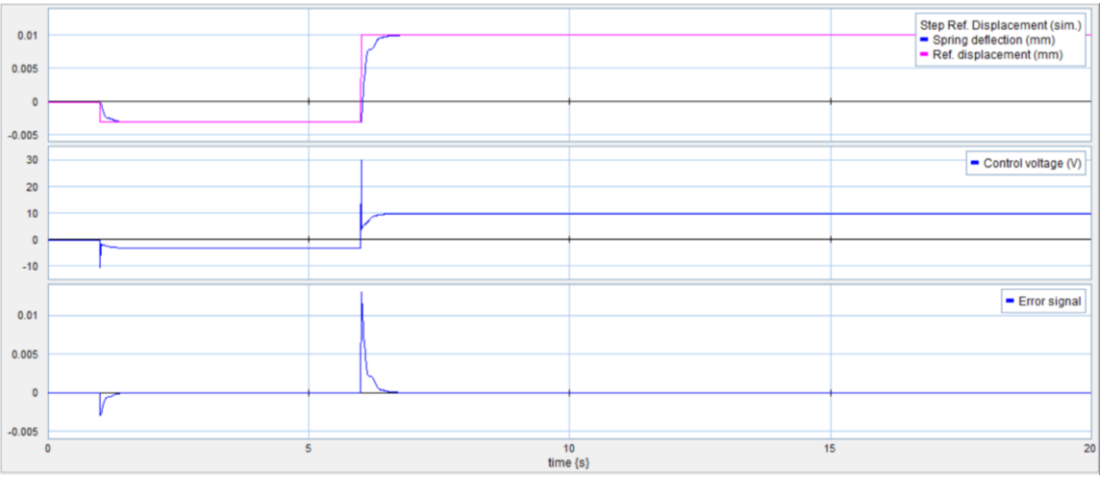


Figure 5.15: Step response of actuator (simulation).

Figure 5.16 shows the experimental result of the same step response. A maximum control voltage of about 15 V is observed. The response follows the step reference signal with a

small overshoot, signifying an underdamped system, and a rise time of about 0.4 seconds. The friction model used in the simulation model is developed for the test rig in its vertical position, while this experiment is done in the horizontal position. Due to the higher normal force of the unsprung mass on the rail in the horizontal position than that in the vertical position, a higher Coulomb friction component is expected in this case, which can most likely be the reason for getting a higher control voltage and rise time.

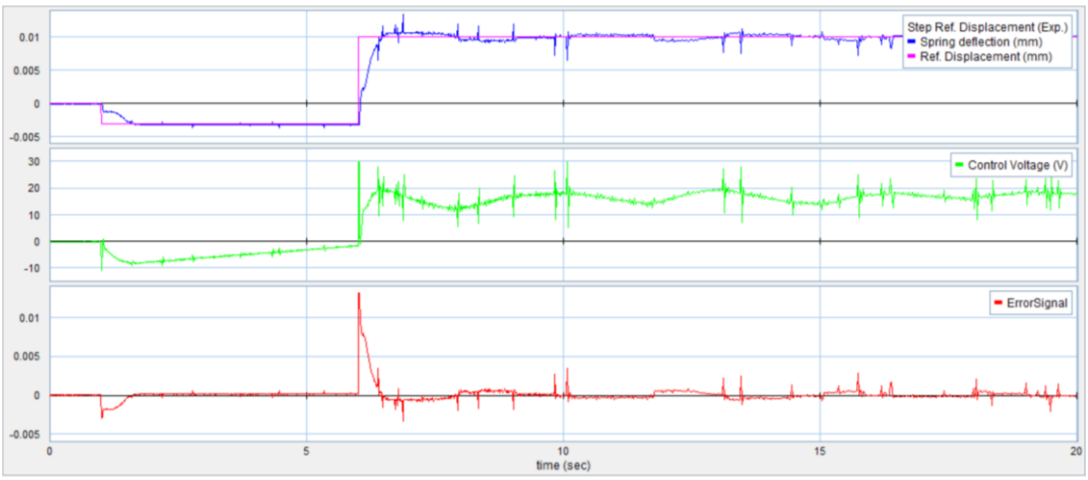


Figure 5.16: Step response of actuator (experiment).

Using the same PID gain values, the model response is tested with respect to a sinusoidal reference displacement signal of 10 mm 0.16 Hz, as shown in figures 5.17 and 5.18. From figure 5.17, the simulation result shows that the model response follows the reference signal with 0.1 second time lag and negligible amplitude loss, for this low frequency reference signal. A peak control voltage of about 9.8 V is observed in the simulation.

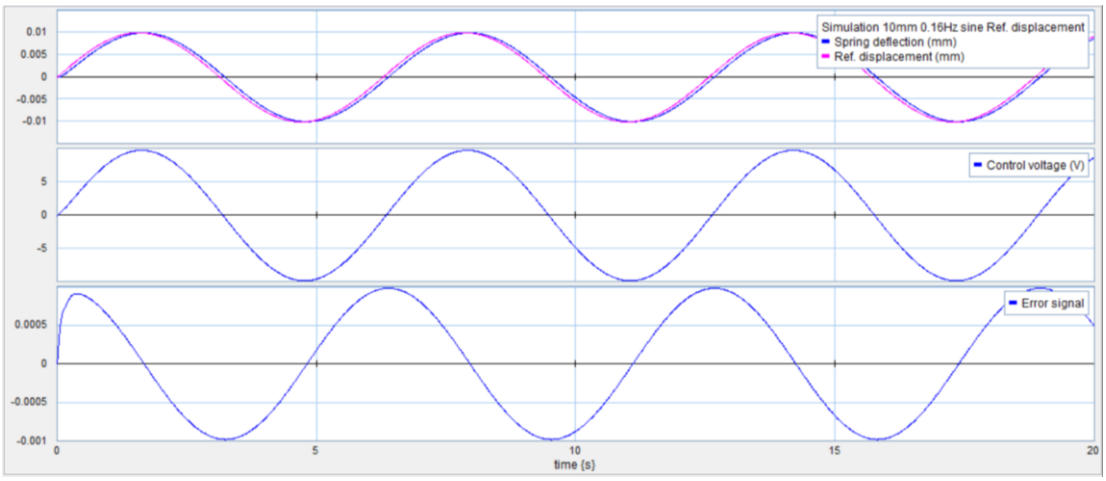


Figure 5.17: Actuator response to 10 mm 0.16 Hz sine reference displacement (simulation).

Figure 5.18 shows the experimental result for the same sine reference signal. The response follows the reference signal almost similarly as the simulation result. The peak control voltage is found to be about 18.5 V. This difference in the values of experimental and simulated control voltage is most likely due to the higher normal force of the unsprung mass on the rail in the horizontal position of the test rig.

Later, a PID gain value of 500, 20000, 100 respectively is found to be more suitable for sinusoidal reference signals in terms of smaller time lag. Similar tests are done using sinusoidal reference signals with higher frequencies, as shown in figures 5.19 – 5.22. More relevant figures are included in appendix E.

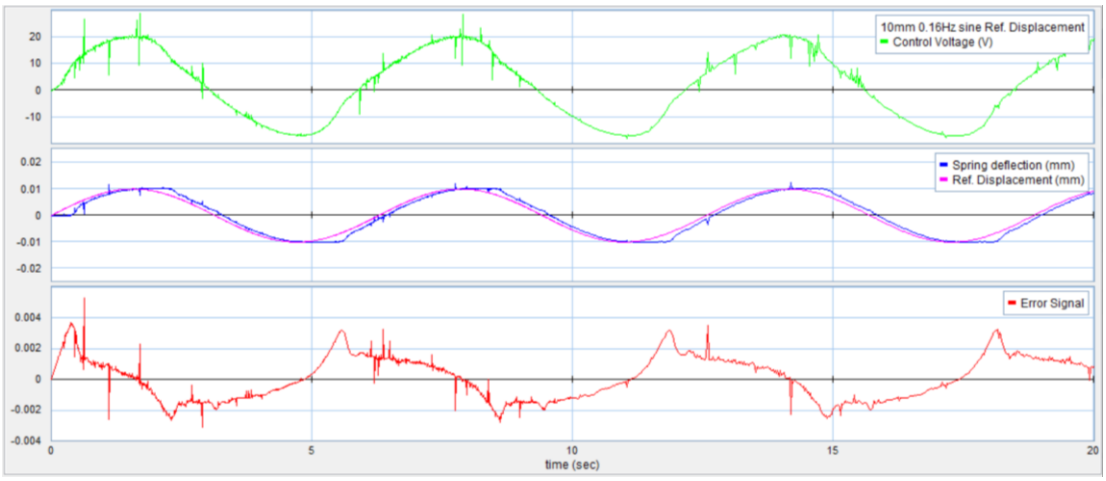


Figure 5.18: Actuator response to 10 mm 0.16 Hz sine reference displacement (experiment).

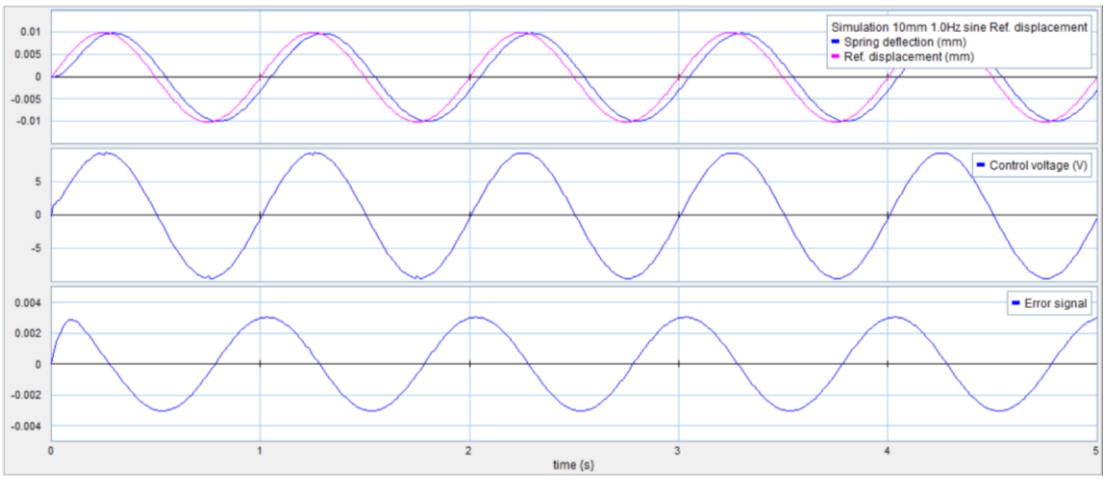


Figure 5.19: Actuator response to 10 mm 1.0 Hz sine reference displacement (simulation).

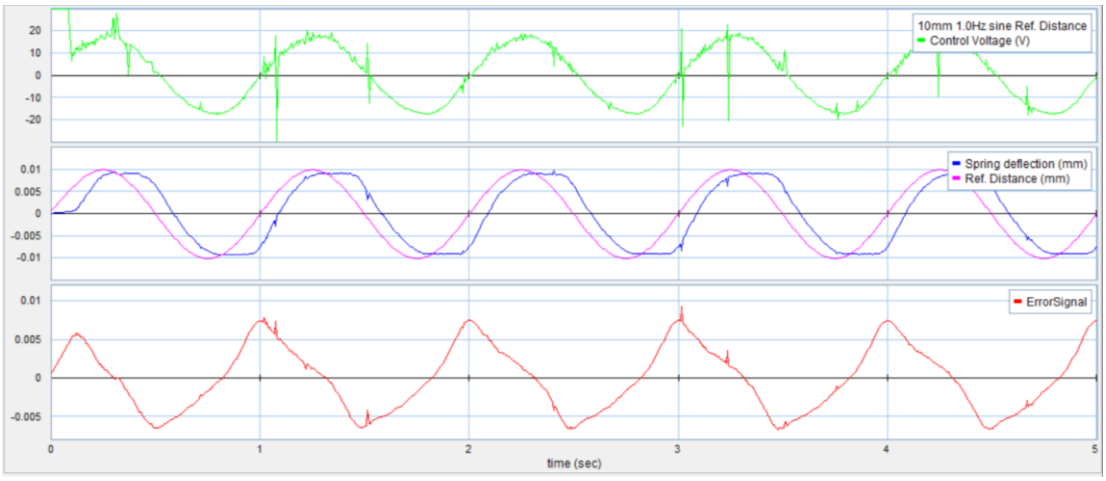


Figure 5.20: Actuator response to 10 mm 1.0 Hz sine reference displacement (experiment).

It is observed that as the frequency of the reference signal increases, the amplitude loss in the response increases. However, the time lag of the response stays within about 0.03 – 0.1 seconds. Up to about 1 Hz frequency, the amplitude loss is negligible (within 10%) during the simulation as well as the experiment, as can be seen in figures 5.19 and 5.20. At 2 Hz frequency, the amplitude loss becomes about 30% during the experiment whereas the simulation predicts an amplitude loss of about 10% (appendix E). Over 5 Hz frequency, the experiment shows an amplitude loss of over 70% whereas the simulation predicts an amplitude loss within 50 – 70%, in all cases, as shown in figures 5.21, 5.22 and appendix E. The higher amplitude loss in the response during the experiment is most likely due to an excessive required force to create the desired acceleration as well as the presence of higher normal force of the unsprung mass on the rail in the horizontal position of the test rig.

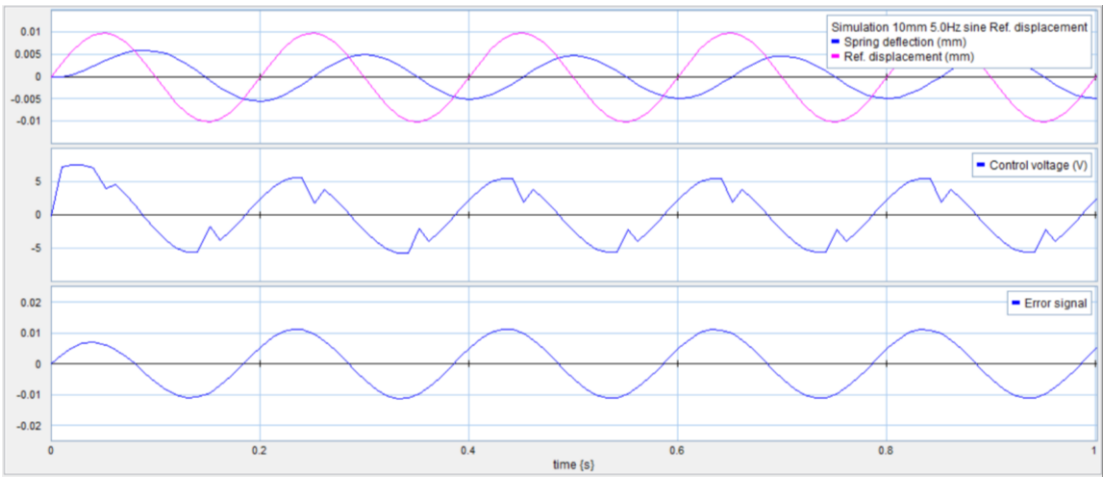


Figure 5.21: Actuator response to 10 mm 5.0 Hz sine reference displacement (simulation).

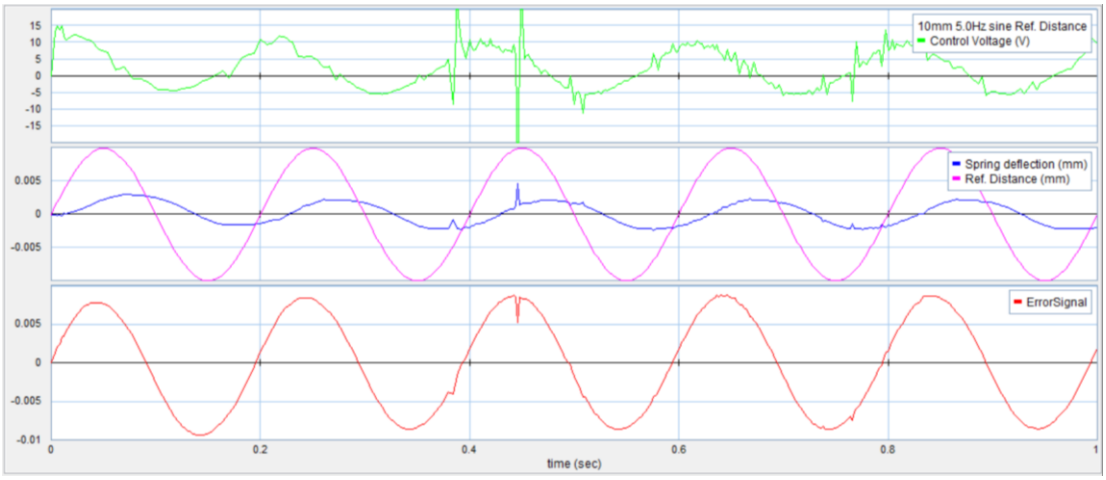


Figure 5.22: Actuator response to 10 mm 5.0 Hz sine reference displacement (experiment).

However, it is more appropriate to control the actuator using a force reference signal. The next section will describe the closed loop force control approach.

5.3.3 Closed Loop Force Control

A closed loop control model is developed to test the actuator response for reference force signal, as shown in figure 5.23. It contains a sub-model 'ControlSystem', as shown in figure 5.24. The sub-model takes in the spring (k_inv) force, i.e., load cell data in terms of voltage (LCvoltage). This voltage data is then filtered and freed from bias using 'Filter_LC' block, and converted back to force value inside the 'SignalMonitor_LC' block. It is then subtracted from the reference force signal (RefForce) to obtain the error signal which passes through the PID blocks (Kp, Ki, Kd). The PID output is kept limited to a safe voltage limit for the actuator inside the 'ControlVoltage' block and passed to the actuator as the control voltage. The actuator gets the control voltage as a PWM signal through the motor driver. In simulation, the control voltage is passed to the 'MSe' element. The 20-sim codes for the model elements are included in the appendix B.

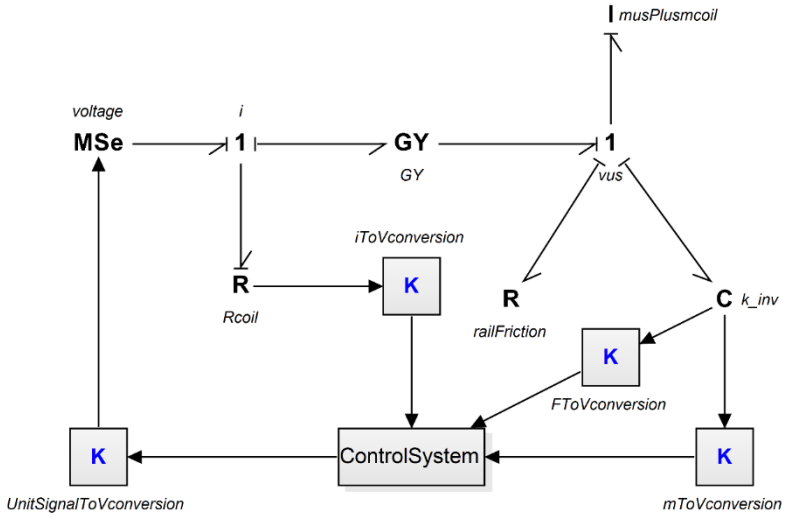


Figure 5.23: Force control model for actuator test with sub-model 'ControlSystem'.

As a first attempt, a 20 N 0.5 Hz sine reference force signal is used. The simulation result for this reference signal is shown in figure 5.25. The force response follows the reference signal with a time lag of about 0.1 seconds and no loss in amplitude. The control voltage is predicted to be about 10 V.

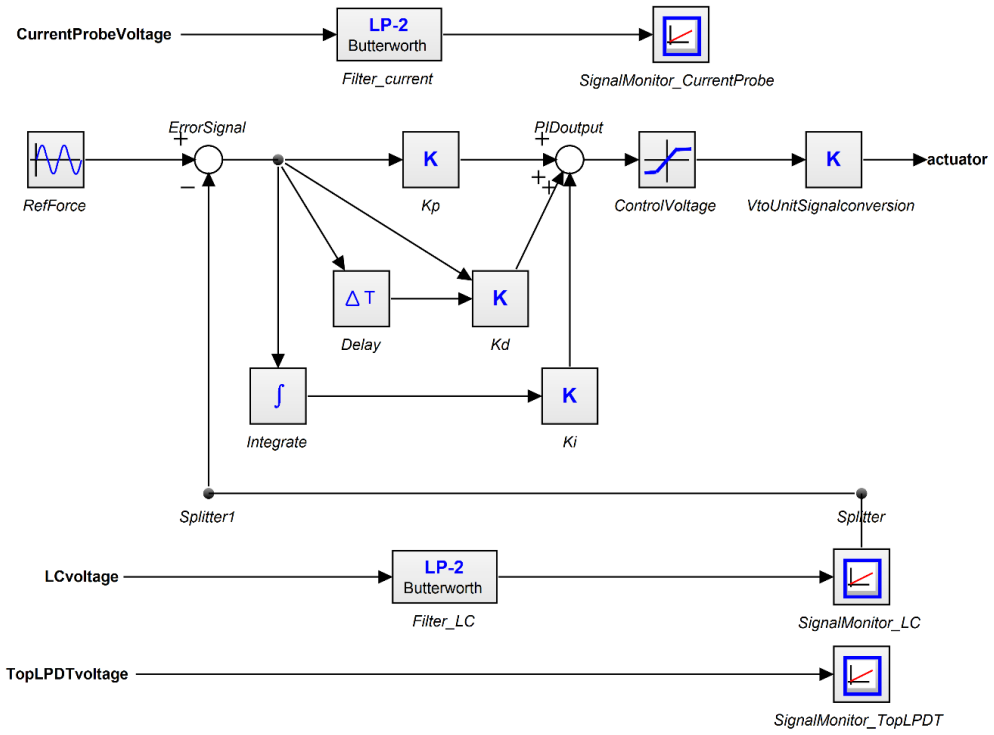


Figure 5.24: Actuator force control sub-model (ControlSystem).

But, from the previous experience of the position control tests, the control voltage is expected to be almost double during the experiment, most likely due to more frictional loss for the unsprung mass on the rail in the horizontal position of the test rig. It is also observed in the figure 5.26. The force response also follows the reference signal expectedly.

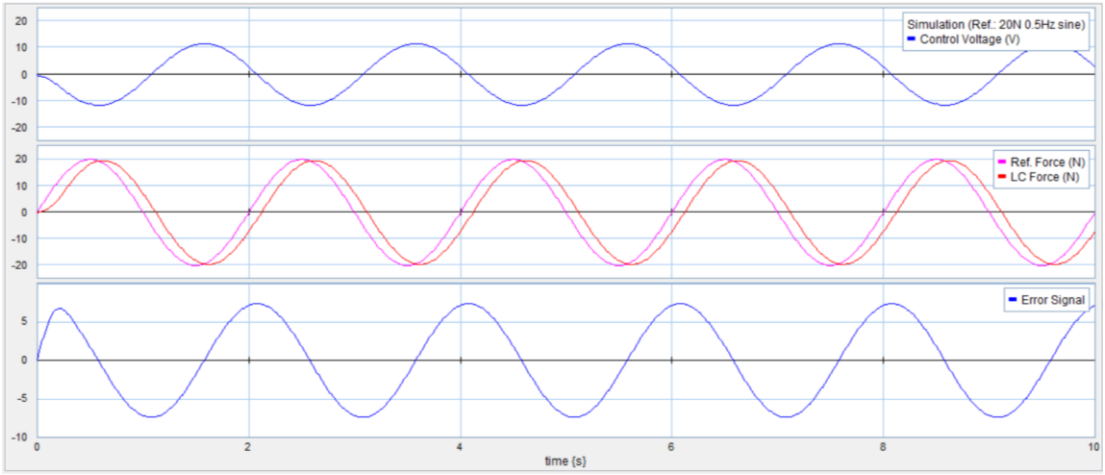


Figure 5.25: Force control test for 20 N 0.5 Hz sine reference force signal (simulation).

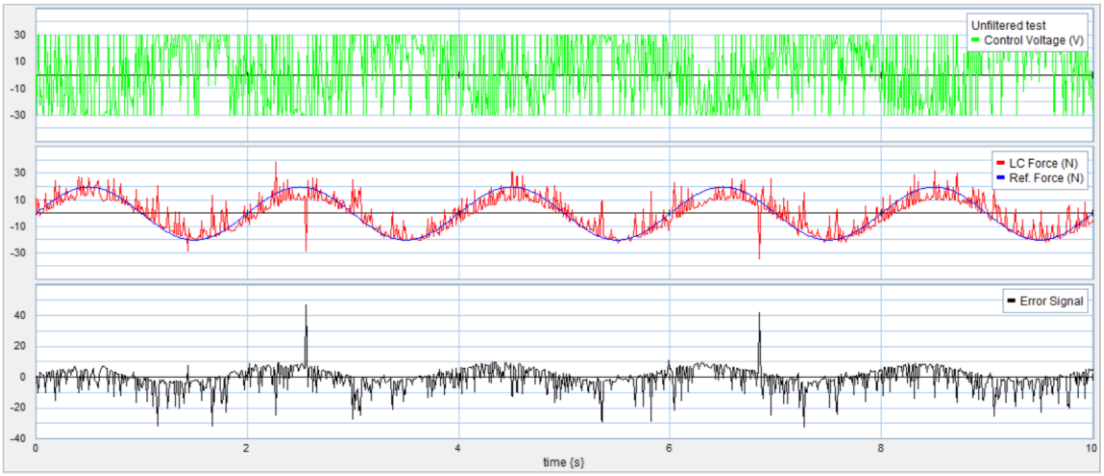


Figure 5.26: Force control test without filter for 20 N 0.5 Hz sine reference force signal (experiment).

However, a loss in amplitude is observed and the load cell output is found to be noisy, as shown in figure 5.27. The amplitude loss is due to the safe limit (±30 V) applied to the

control voltage for the actuator. Since the full scale of the load cell used is ±250 lb, it is expected to be noisy with this very small force application.

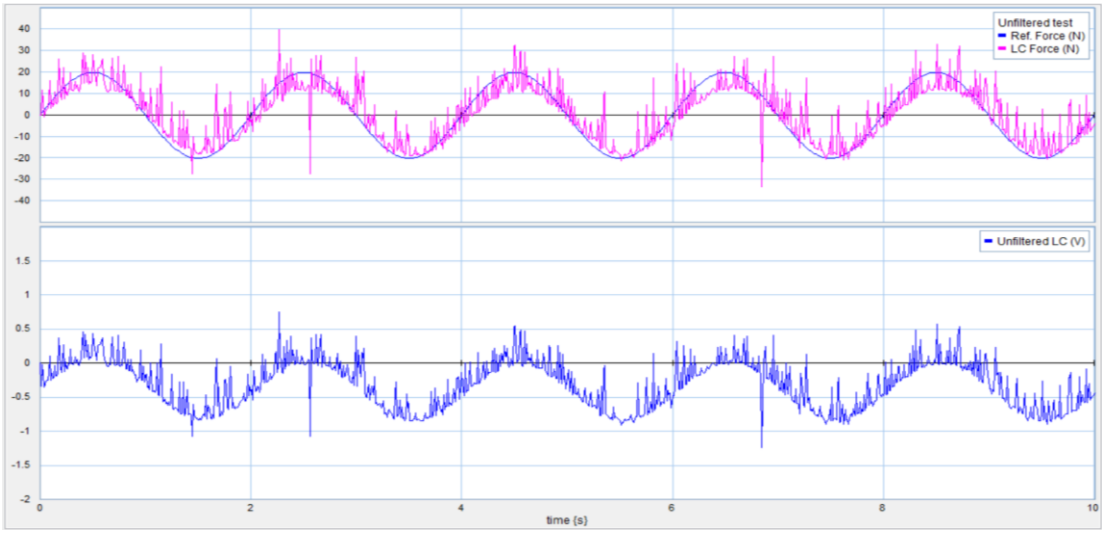


Figure 5.27: Unfiltered load cell data (experiment).

An FFT analysis is done for the load cell voltage output of 0.5 Hz sine, as shown in the figure 5.28. The Matlab code for the FFT analysis is given in the appendix C. The FFT plot shows that noises of very small and almost equal amplitude are present over a wide range of frequencies. Some unwanted frequency contents of significantly visible amplitudes are observed below 5 Hz frequency.

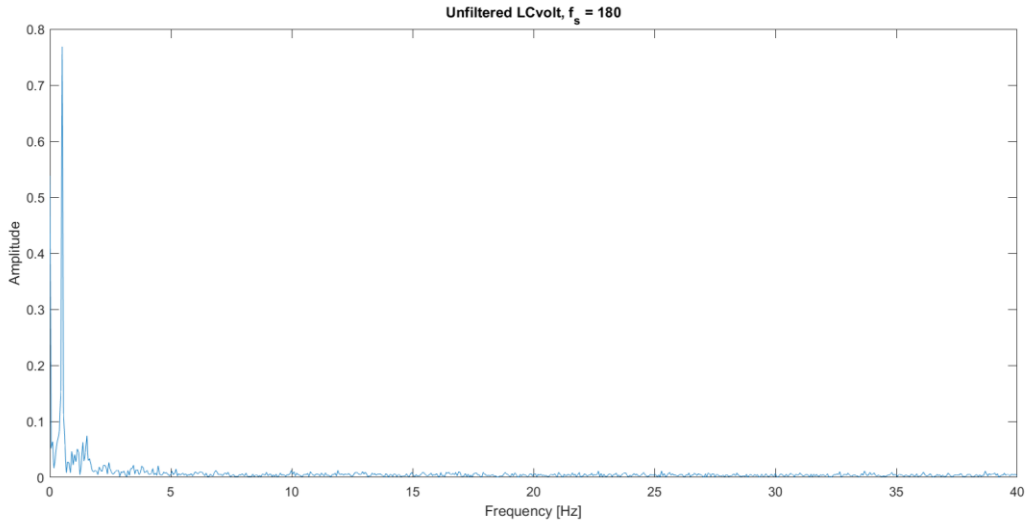


Figure 5.28: FFT analysis of the load cell voltage (experiment).

By trial and error, a 2nd order 3 Hz low-pass Butterworth filter is found to be suitable for this application. The associated plots are given in the appendix E. A similar filter of 20 Hz is not that effective, and of 10 Hz is quite effective. However, this restriction of frequency will limit the application of this load cell for this kind of purpose; a load cell of about ±100 N full-scale is recommended. No significant amplitude loss is observed for these filters. However, the 3 Hz filter introduces a time lag of about 0.07 seconds. The amplitude loss observed in the force response is due to the control voltage limit.

After using the 3 Hz filter, the load cell signal becomes noise free, as shown in figure 5.29, compared to that of the figure 5.27. Since the control voltage is about ±18 V, i.e., within the safe limit, no amplitude loss in the force response is observed, as can be seen in the figure 5.30. The time lag is slightly increased to about 0.15 seconds due to the filter.

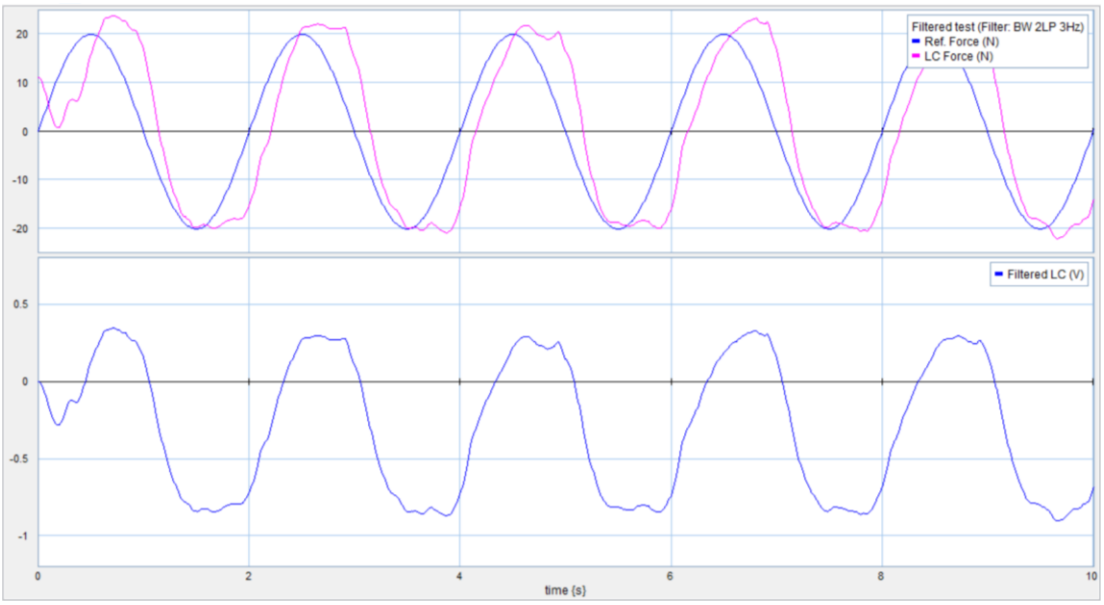


Figure 5.29: Filtered load cell data (experiment).

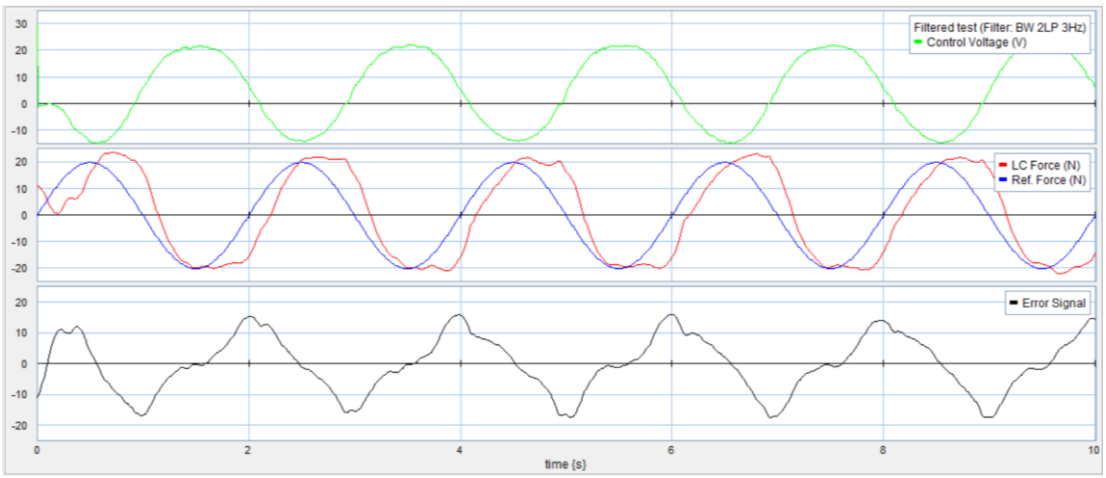


Figure 5.30: Force control test with filter for 20 N 0.5 Hz sine reference force signal (experiment).

Another force control test is conducted for a car hitting a 30 cm long and 2 cm high road bump at 72 km/h speed, as shown in figures 5.31 and 5.32. The actuator force generated

using the active model corresponding to this road bump is used as the reference force signal. No filter is used, as the frequency of the reference signal is about 11 Hz. The force response follows the reference signal with a time lag of about 0.05 seconds and a significant loss in amplitude. Again, the amplitude loss happens mainly due to the safe limit of the control voltage. Similar results can be found using a similar pothole instead of a road bump. The associated plots are given in the appendix E.

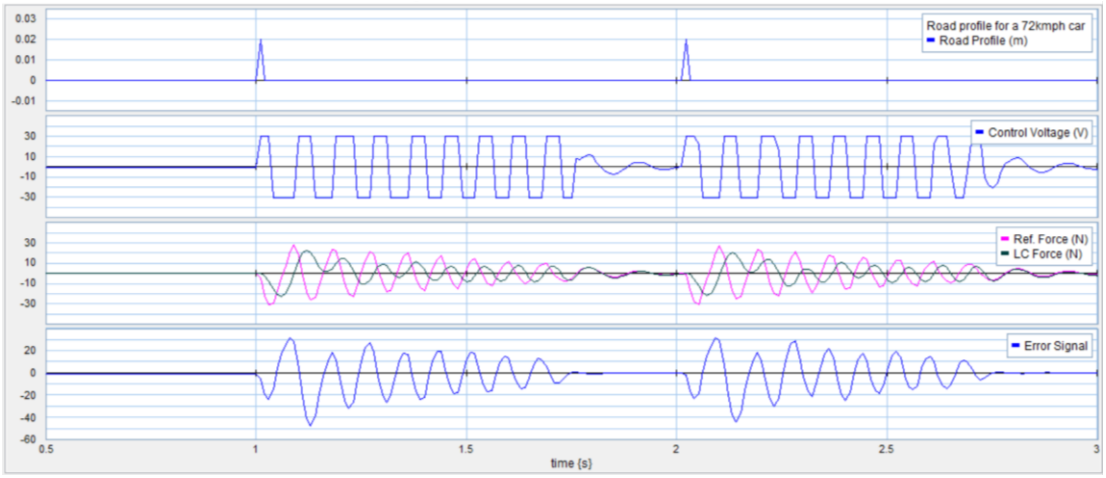


Figure 5.31: Force control test without filter for a car hitting a 30 cm long and 2 cm high road bump at 72 km/h speed (simulation).

According to the above analyses, the linear motor can generate about 20 N force when the reference force frequency is low (figure 5.30), but it can generate only about 5 N force when the reference force frequency is above about 10 Hz (figure 5.32). Hence, the linear motor is not ready to be used in the quarter car test rig as an effective means of actuation. A load cell with smaller full-scale range (about ±100 N) is also recommended for further experimentation.

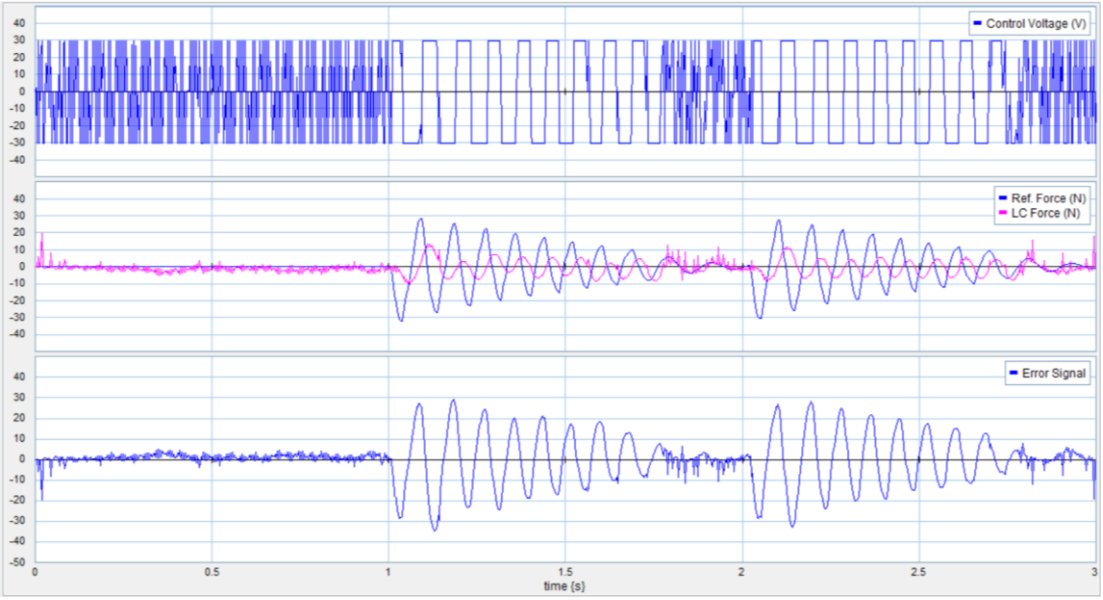


Figure 5.32: Force control test without filter for a car hitting a 30 cm long and 2 cm high road bump at 72 km/h speed (experiment).

However, the above experiments and the control models show the essential interfacing between software and hardware. The chapter explains how a simple PID controller designed and implemented in 20-sim software can be used to control a piece of hardware by reading in sensor signals, generating error signals, and then creating a control voltage signal. Some lags are expected in the hardware system, when it will be reassembled into a fully-functioning quarter car with active suspension.

The next chapter will enhance the quarter car model in 20-sim from chapter 4 by including actuator dynamics and a second feedback control loop for force control. It will compare the behaviours of ideal active control, and realistic active control with the actuator dynamics and additional force control loop.

Chapter 6: Comparative Study of Ideal and Realistic Active Control Model

An idealized controller designed based on a linear model of the test rig in chapter 4 was used to develop an ideal active control model in 20-sim. That model didn't include the actuator dynamics. During hardware experiments, a linear motor will be used as a means of actuation, which will be controlled using force control signal. A load cell placed in the setup will measure the actuator force, and this force signal will be used as a feedback signal for the active control model. Hence, it is necessary to enhance the control model in 20-sim by introducing the actuator dynamics and an additional force control loop to make the model realistic. The chapter will develop the realistic model, and compare the performances of the ideal and realistic active control models.

6.1 Ideal Active Control Model

Based on the analyses described in chapter 3 and 4, the nonlinear passive and active control model for the test rig was developed. A simultaneous simulation model for both passive and active models is shown in figure 6.1. The models are said to be ideal as the actuator dynamics and load cell equivalent elements are not present, instead the active force is directly applied to the model using a modulated effort source 'MSe' element (actuator).

Different road profile velocity signals are applied to a splitter (vr) which allows the signals to be applied to both models simultaneously through two modulated flow source 'MSf' elements. The MSf supplies this flow to the tire. The tire capacitance (kt) stores some energy and creates a velocity difference between the road and the unsprung mass. The rail friction (Rus) dissipates some energy from the system due to the unsprung mass motion on the rail. The unsprung mass (mu) includes the actuator coil mass. It shows some inertial effect and includes some effort into the system due to its weight (Wu). Then the suspension capacitance (ksusp) stores some energy, and the suspension damping (bsusp) dissipates some energy from the system. The suspension creates a velocity difference between the unsprung and sprung masses. The sprung mass (ms) includes the actuator body mass. Again, the rail friction (Rs) dissipates some energy due to the sprung mass motion on the rail. The sprung mass has some inertial effect on the system and includes some effort into the system due to its weight (Ws).

In the active model however, the four system states – suspension deflection (zsusp1), sprung mass velocity (vs1), tire deflection (zt1), unsprung mass velocity (vus1), are taken into a multiplexer element (Mux) to supply these states to the MSe element (actuator) as a single state matrix. Inside the MSe element, this matrix is then multiplied with the feedback gain matrix G (calculated in chapter 4) to obtain the required actuator force. This force is then applied to the model to control the suspension velocity (vsusp1) and attenuate the three transfer functions, given in equations 3.19 – 3.21. The 20-sim codes for the bond graph elements are given in appendix B.

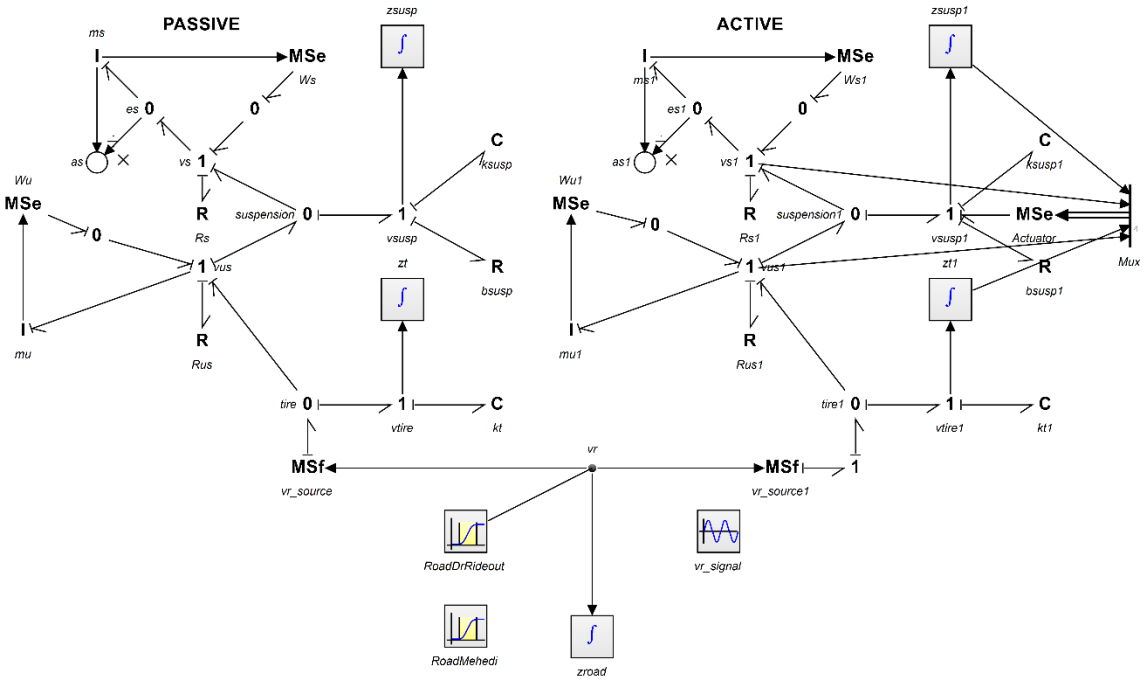


Figure 6.1: Ideal passive and active control model.

6.2 Realistic Active Control Model

A more realistic version of the previous model (figure 6.1) is developed by introducing the actuator dynamics and an additional force control loop, as shown in figures 6.2 and 6.3. In the test setup, the load cell is placed in between the actuator coil mass and the unsprung mass to measure the actuator force. Hence, in this model, the unsprung mass (mus) doesn't include the actuator coil mass (mcoil), and a very high stiffness spring (kLC) is inserted in between these two masses to represent the load cell (figure 6.2). The very small load cell capacitance creates a very small velocity difference between the unsprung and actuator coil masses.

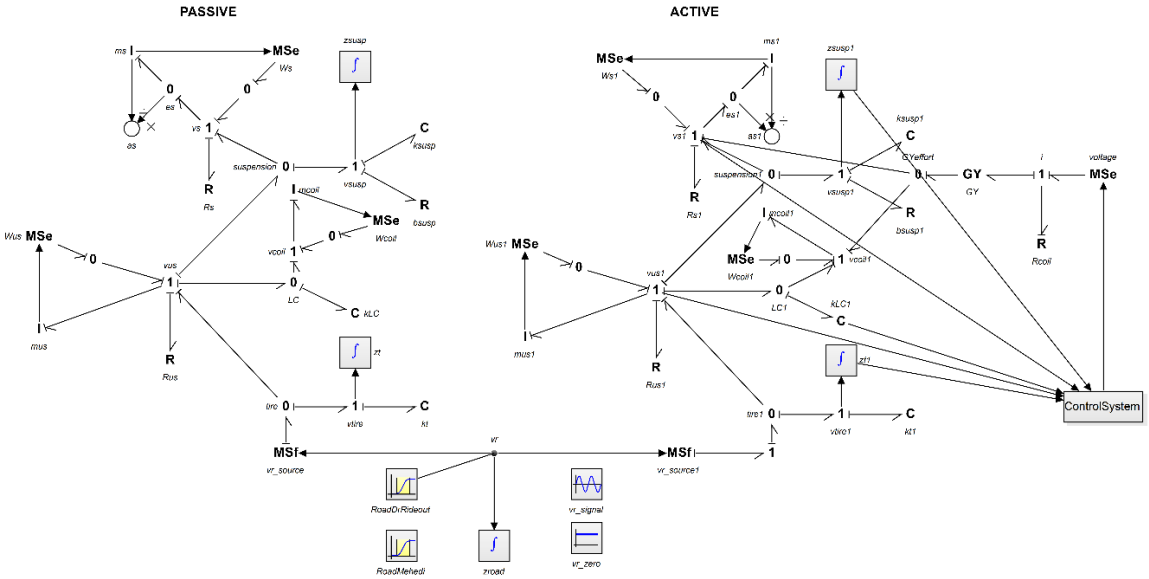


Figure 6.2: Realistic passive and active control model with sub-model 'ControlSystem'.

The control operations are done inside the sub-model 'ControlSystem', as shown in figure 6.3. This sub-model will only be exported to 20-sim 4C as C code for the experiment. It takes in the four system states and load cell effort as sensor signals to do the control operations. Inside the sub-model, the four states (zsusp, zs_dot, ztire, zus_dot) are converted into a single state matrix using a multiplexer element (StateMatrix) and supplied to the 'RefForce' block, where this matrix is then multiplied with the feedback gain matrix G (calculated in chapter 4) to obtain the required actuator force as the reference force signal. The load cell force (Flc) is freed from its initial reading (Flc_initial) and then subtracted from the reference force to calculate the error signal (ErrorSignal_F). The error signal then passes through the PID blocks (Kp, Ki, Kd) to calculate the required control voltage. The PID values used are 10, 5, 0.001 respectively.

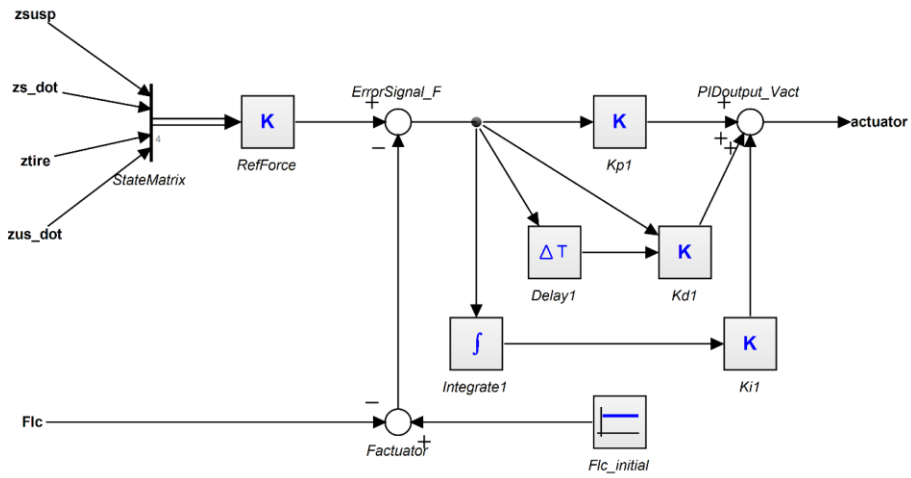


Figure 6.3: Sub-model 'ControlSystem'.

In figure 6.2, the control voltage is then supplied to the actuator, i.e., the MSe element (voltage) and converted into the actuator force using a gyrator element (GY). The force is then applied on the sprung and actuator coil mass to attenuate the three transfer functions, given in equations 3.19 – 3.21. The 20-sim codes for the bond graph elements are given in appendix B.

6.3 Performance Comparison

6.3.1 Road Bump

A road profile equivalent to a car passing a 30 cm long and 2 cm high road bump at 72 km/h is used to test these two models, as shown in figures 6.4 and 6.5 respectively. Comparing these two figures, a peak required actuator force of about 24.5 N is observed for both models, which is achievable using the available linear motor. The realistic model shows, using the second force control loop, the actuator can generate a peak force of about 18 N with a very negligible time lag, and it requires a peak control voltage of about 13.5 V.

In the ideal model, the idealized controller (developed based on a linear test rig model) slightly increases the sprung mass acceleration at the beginning and then eventually reduces it, whereas the realistic active model is more successful in reducing the sprung mass acceleration and it also reduces sprung mass oscillation. However, the ideal active model shows slightly better suspension deflection and road holding performance compared to that of the realistic active model.

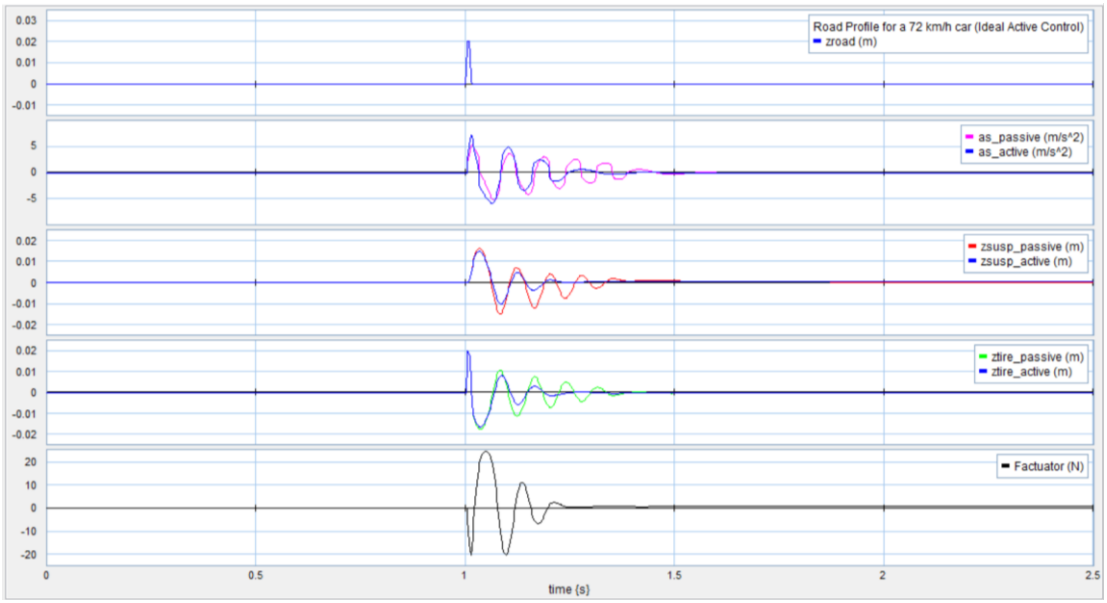


Figure 6.4: Ideal active control for a car hitting a 30 cm long and 2 cm high road bump at 72 km/h speed.

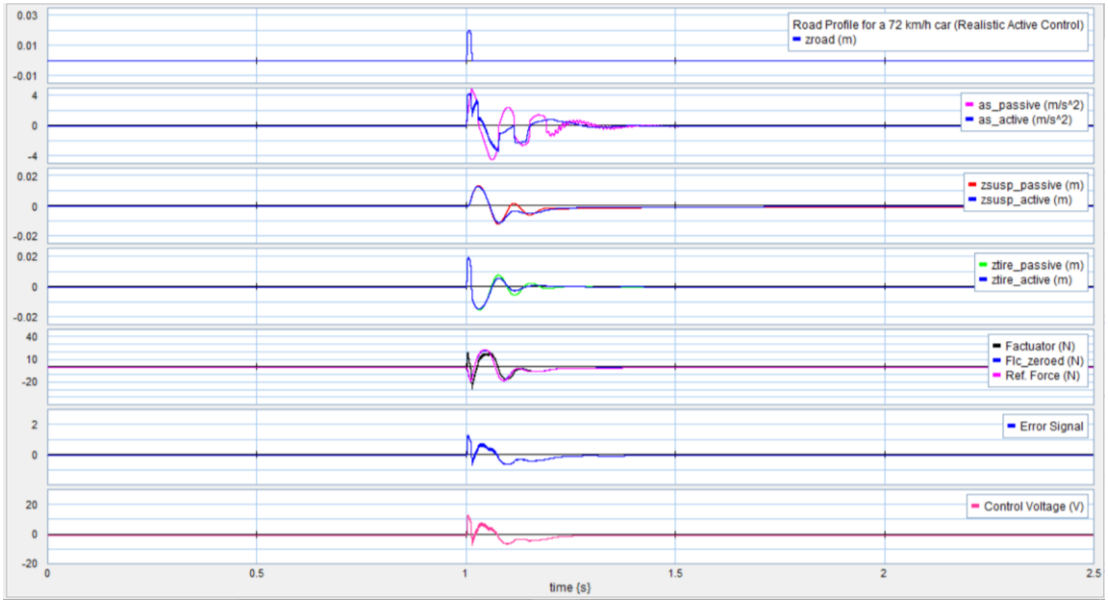


Figure 6.5: Realistic active control for a car hitting a 30 cm long and 2 cm high road bump at 72 km/h speed.

6.3.2 Sinusoidal Road Input

A high frequency sine (2 mm 15 Hz sine) road profile is used to test both models, as shown in figures 6.6 and 6.7 respectively. A peak required actuator force of about 9.9 N for the ideal model and of about 6.4 N for the realistic model are observed. From figure 6.7, it is found that the actuator can generate a peak force of about 6 N for the realistic model with a negligible time lag of about 0.006 seconds, and it requires a peak control voltage of about 1.74 V.

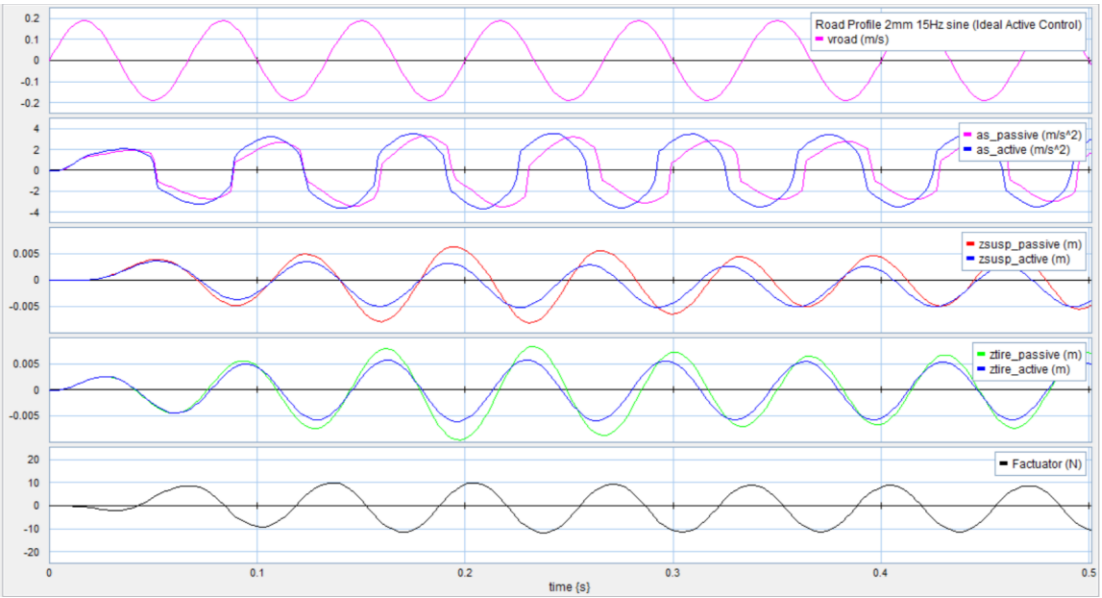


Figure 6.6: Ideal active control for a 2 mm 15 Hz sine road input.

The ideal active model is slightly making things worse for the sprung mass acceleration. The realistic active model shows better performance in reducing the sprung mass acceleration compared to that of the ideal active model. The ideal active model shows

slightly better suspension deflection and road holding performance compared to that of the realistic active model.

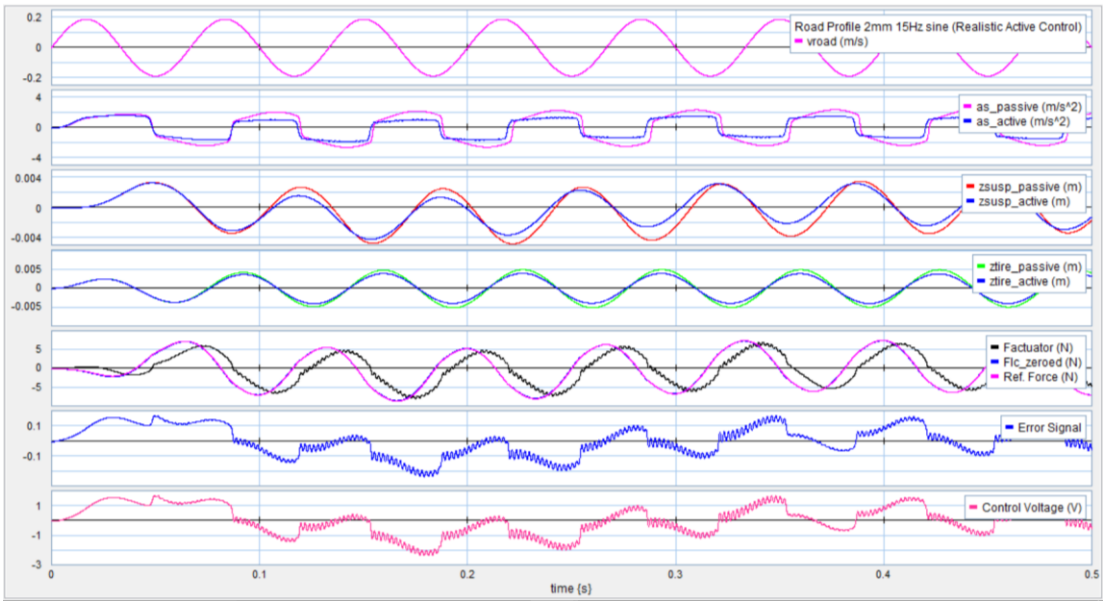


Figure 6.7: Realistic active control for a 2 mm 15 Hz sine road input.

Again, a low frequency sine (10 mm 0.5 Hz sine) road profile is used to test both models, as shown in figures 6.8 and 6.9 respectively. A peak required actuator force of about 1.45 N for the ideal model and of about 0.56 N for the realistic model are observed. From figure 6.9, it is found that the actuator can generate a peak force of about 0.5 N for the realistic model with a very negligible time lag, and it requires a peak control voltage of about 0.9 V.

The effect on the sprung mass acceleration is very negligible for both models. The ideal active model decreases the suspension deflection during the negative cycle and increases it during the positive cycle, and the model doesn't enhance the road holding performance.

The realistic active model however slightly enhances the suspension deflection and road holding performance.

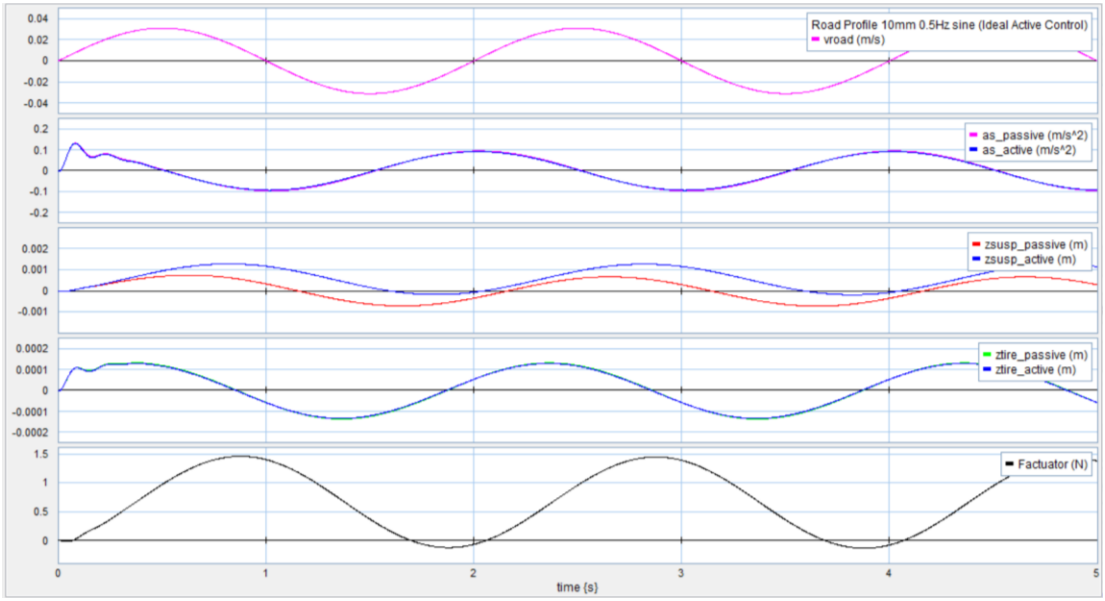


Figure 6.8: Ideal active control for a 10 mm 0.5 Hz sine road input.

Based on the above analyses, the realistic active model generally performs better to attenuate the sprung mass acceleration, specially when the road input has high frequency, compared to the ideal active model. On the other hand, the ideal active model generally performs slightly better to attenuate the suspension and tire deflections compared to the realistic active model. However, when the road input has low frequency, the realistic active model performs slightly better in reducing the suspension and tire deflections.

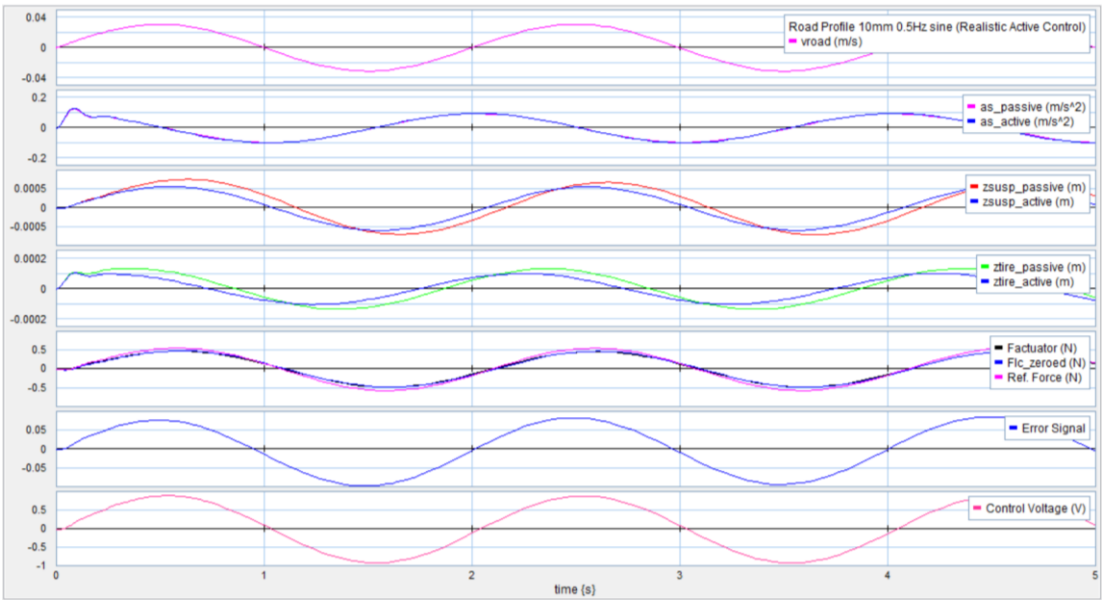


Figure 6.9: Realistic active control for a 10 mm 0.5 Hz sine road input.

The realistic model represents the quarter car test rig completely. It shows how the control sub-model 'ControlSystem' takes in the sensor signals, generates error signal and creates a control voltage signal in 20-sim 4C, to control the test rig. However, the model will require some minor modifications to be used in the experiment, depending on the calibration equations of the sensors used. During the experiment, the sub-model will read in the sensor signals in terms of voltage, so these signals need to be converted back into the corresponding units inside the sub-model, using the calibration equations of the sensors.

Chapter 7: Conclusion

7.1 Summary

In this thesis, the parameters of an existing quarter car test rig are estimated, and control infrastructure is developed to be used in future to do quarter car performance tests. Different static and dynamic tests are done to estimate the test rig parameters – suspension and tire stiffness, suspension damping and rail frictions corresponding to sprung and unsprung masses. Suspension and tire spring stiffness are measured using static compressive load tests. All springs are found to be very linear in nature. The suspension damping and the rail frictions for the two masses are found by cycling the individual mass using some dynamic force against a spring and/or a damper, and free fall test of the unsprung mass, along the rail. The corresponding force and mass motion data are measured by load cell, accelerometer and LPDT. A passive model for the test rig is then developed using these estimated parameters and nonlinearities.

For simplicity, a similar model neglecting the nonlinearities in the parameters, i.e., a linear test rig model is used to design an idealized LQR based controller. Then, the passive

nonlinear model is modified by introducing this idealized controller to make the model active. This active model is ideal in the sense that it doesn't include the actuator dynamics. The idealized controller performance is tested and compared for both the linear and nonlinear active models. Its performance is found to be slightly better at low frequencies up to the sprung mass resonant frequency for the nonlinear model compared to that of the linear model. The controller performance is also tested using sinusoidal road inputs. It is found that the nonlinearities present in the suspension damper degrades the controller performance for low frequency road inputs.

Before introducing a linear motor in the test setup as a means of actuation, its static and dynamic responses are tested using different open loop, and closed loop position and force control tests. It is found that the linear motor can generate about 20 N force when the frequency of the reference force signal is low, but it can generate only about 5 N force when the reference force signal frequency is above about 10 Hz. Hence, this linear motor wouldn't be an effective means of actuation. However, the experiments and the control models show the essential interfacing between software and hardware.

The actuator dynamics is then introduced in the ideal active control model developed earlier to make it realistic. The performance of the ideal and realistic active control models is then compared. It is found that the realistic active control model generally performs better in reducing the sprung mass acceleration specially when the road input has high frequency, and in reducing the suspension and tire deflections when the road input has low frequency.

The realistic control model will show the future researchers the complete representation of the test rig, and how 20-sim 4C uses the model to read in the sensor signals, generate error signal and create a control voltage signal.

7.2 Recommendations and Future Works

The realistic control model will be implemented using 20-sim 4C in the test rig to do real control tests with the complete quarter car. The model will require some minor modifications to be able to convert the sensor voltage signals into the corresponding units using the calibration equations of the sensors, during the experiment. The simulation model can be enhanced to test situations like a continuous uphill or downhill road profile, and by limiting the suspension and tire deflections to test all kinds of road profile scenarios. However, before doing the experiments, some things may need to be taken care of -

- A linear shock absorber is recommended, since its nonlinearity degrades the idealized controller performance for low frequency road inputs.
- A linear motor with higher continuous force and power rating is recommended, since the current linear motor is unable to generate sufficient force for high frequency reference force signals.
- A load cell with smaller full-scale range (about ±100 N) is recommended, since the full-scale range (±250 lbs) of the current load cell is too high for these experiments.

Future experiments can be done with the existing actuator, using a very simple control scheme such as direct feedback control of sprung mass acceleration. That would be a very simple test with one sensor, which might reveal benefits of active suspension control even if the theoretically ideal actuator force can not be achieved at moderate to high frequencies.

Bibliography

[1] T. D. Gillespie, Fundamentals of Vehicle Dynamics, Society of Automotive Engineers, Inc.

[2] R. Rajamani, Vehicle Dynamics and Control, Springer, 2012.

[3] D. Hrovat, "Applications of Optimal Control to Advanced Automotive Suspension Design," *Journal of Dynamic Systems, Measurement and Control,* vol. 115, no. 2B, pp. 328-342, 1993.

[4] T. Dahlberg, "Optimization Criteria for Vehicles Travelling on a Randomly Profiled Road - a Survey," *Vehicle System Dynamics,* vol. 8, no. 4, pp. 239-352, 1979.

[5] Ride and Vibration Data Manual, SAE J6a, Society of Automotive Engineers, 1965.

[6] D. C. Karnopp, D. L. Margolis and R. C. Rosenberg, "Chapter 2: Multiport Systems and Bond Graphs," in *System Dynamics: Modeling, Simulation and Control of Mechatronic Systems*, New Jersey, John Wiley & Sons, Inc., 2012.

[7] P. C. Breedveld and H. Unbehauen, "Modeling and Simulation of Dynamic Systems using Bond Graphs," in *Control Systems, Robotics and Automation*, Oxford, Encyclopedia of Life Support Systems, 2014.

[8] "20-sim Factsheet," Controllab Products B.V., www.20sim.com.

[9] S. Brennan and A. Alleyne, "A Scaled Testbed for Vehicle Control: the IRS," *Proceedings of the 1999 IEEE International Conference on Control Applications,* vol. 1, pp. 327-332, 1999.

[10] S. Brennan and A. Alleyne, "Using a Scale Testbed: Controller Design and Evaluation," *IEEE Control Systems,* vol. 21, no. 3, pp. 15-26, 2001.

[11] M. D. Petersheim and S. N. Brennan, "Scaling of Hybrid-Electric Vehicle Powertrain Components for Hardware-in-the-Loop Simulation," *Mechatronics,* vol. 19, no. 7, p. 1078–1090, 2009.

[12] J. Langdon and S. C. Southward, "Development of a General Use Quarter-Vehicle Test Rig," *Proceedings of the ASME 2007 International Design Engineering Technical*

Conferences & Computers and Information in Engineering Conference, vol. 3, pp. 1239-1245, 2007.

[13] T. Tseng and D. Hrovat, "Some Characteristics of Optimal Vehicle Suspensions Based on Quarter-Car Models," *Proceedings of the 29th IEEE Conference on Decision and Control,* vol. 4, pp. 2232-2237, 1990.

[14] W.-J. Evers, I. Besselink, A. Teerhuis, T. Oomen and H. Nijmeijer, "Experimental Validation of a Quarter Truck Model Using Asynchronous Measurements with Low Signal-to-Noise Ratios," *Proceedings of the 10th International Symposium on Advanced Vehicle Control,* pp. 177-182, 2010.

[15] W.-J. Evers, I. Besselink, A. Teerhuis, T. Oomen and H. Nijmeijer, "Experimental Validation of a Truck Roll Model Using Asynchronous Measurements with Low Signal-to-Noise Ratios," in *American Control Conference*, Baltimore, 2010.

[16] M. I. Ahmed, H. M. Yusof and M. M. Rashid, "Modeling a Small-Scale Test Rig of Quarter Car Railway Vehicle Suspension System," *International Journal of Intelligent Mechatronics and Robotics,* vol. 2, no. 4, pp. 149-153, 2016.

[17] G. Koch, E. Pellegrini, S. Spirk and B. Lohmann, "Design and Modeling of a Quarter-Vehicle Test Rig for Active Suspension Control," Technical Reports on Automatic Control, Garching, 2010.

[18] Y. Taskin, N. Yagiz and I. Yuksek, "Lumped Parameter Identification of a Quarter Car Test Rig," in *International Conference on Mathematical Modelling in Physical Sciences*, 2013.

[19] C. Sandu, E. R. Andersen and S. Southward, "Multibody Dynamics Modelling and System Identification of a Quarter-Car Test Rig with McPherson Strut Suspension," *Vehicle System Dynamics,* vol. 49, no. 1-2, p. 153–179, 2011.

[20] M. K. Salaani, D. A. Guenther and G. J. Heydinger, "Vehicle Dynamics Modeling for the National Advanced Driving Simulator of a 1997 Jeep Cherokee," SAE Technical Paper, 1999.

[21] G. J. Heydinger, M. K. Salaani, W. R. Garrott and P. A. Grygier, "Vehicle Dynamics Modelling for the National Advanced Driving Simulator," *Journal of Automobile Engineering,* vol. 216, no. 4, pp. 307-318, 2002.

[22] M. K. Salaani, G. J. Heydinger and P. A. Grygier, "Parameter Determination and Vehicle Dynamics Modeling for the NADS of the 1998 Chevrolet Malibu," SAE Technical Paper, 2001.

[23] "NADS Vehicle Dynamics Simulation, Release 4.0," Center for Computer-Aided Design, National Advanced Driving Simulator, University of Iowa, Iowa, 1994.

[24] D. Karnopp, "Theoretical Limitations in Active Vehicle Suspensions," *Vehicle System Dynamics,* vol. 15, no. 1, pp. 41-54, 1986.

[25] G. Verros, S. Natsiavas and C. Papadimitriou, "Design Optimization of Quarter-Car Models with Passive and Semi-Active Suspensions under Random Road Excitation," *Journal of Vibration and Control,* vol. 11, pp. 581-606, 2005.

[26] D. Hrovat, "Survey of Advanced Suspension Developments and Related Optimal Control Applications," *Automatica,* vol. 33, no. 10, p. 1781–1817, 1997.

[27] C. Lauwerys, J. Swevers and P. Sas, "Design and Experimental Validation of a Linear Robust Controller for an Active Suspension of a Quarter Car," in *American Control Conference*, Boston, 2004.

[28] C. Lauwerys, J. Swevers and P. Sas, "Robust Linear Control of an Active Suspension on a Quarter Car Test-Rig," *Control Engineering Practice,* vol. 13, no. 5, p. 577–586, 2005.

[29] C. McGinn and D. Geraghty, "Modelling an Active Suspension Controller for a Road Vehicle," in *Signals and Systems Conference (ISSC 2010), IET Irish*, Cork, 2010.

[30] S. Chantranuwathana and H. Peng, "Adaptive Robust Control for Active Suspensions," in *American Control Conference*, San Diego, 1999.

[31] A. Alleyne and J. K. Hedrick, "Nonlinear Adaptive Control of Active Suspensions," *IEEE Transactions on Control Systems Technology,* vol. 3, no. 1, pp. 94-101, 1995.

[32] D. G. Rideout and K. T. Haq, "Active Modeling: a Method for Creating and Simulating Variable-Complexity Models," *Journal of Dynamic System, Measurement and Control,* vol. 132, no. 6, 2010.

[33] D. G. Rideout, "Simulating Coupled Longitudinal, Pitch and Bounce Dynamics of Trucks with Flexible Frames.," *Modern Mechanical Engineering,* vol. 2, pp. 176-189, 2012.

[34] D. Margolis and T. Shim, "A Bond Graph Model Incorporating Sensors, Actuators and Vehicle Dynamics for Developing Controllers for Vehicle Safety," *Journal of the Franklin Institute,* vol. 338, no. 1, pp. 21-34, 2001.

[35] J. Deur, M. Cipek and J. Petric, "Bond Graph Modeling of Series-Parallel Hybrid Electric Vehicle Power Train Dynamics," in *SpringSim '10, Spring Simulation Multiconference*, Orlando, 2010.

[36] N. Banerjee, A. K. Saha, R. Karmakar and R. Bhattacharyya, "Bond Graph Modeling of a Railway Truck on Curved Track," *Modelling Practice and Theory,* vol. 17, no. 1, pp. 22-34, 2009.

[37] H. Adibi-asl and D. G. Rideout, "Bond Graph Modeling and Simulation of a Full Car Model with Active Suspension," Faculty of Engineering, Memorial University of Newfoundland, St. John's.

[38] K. J. Wakeham, "Development of an Active Suspension Scale Vehicle Platform," Department of Engineering, Memorial University of Newfoundland.

[39] D. G. Rideout, J. L. Stein and L. S. Louca, "Systematic Identification of Decoupling in Dynamic System Models," *Journal of Dynamic Systems, Measurement and Control,* vol. 129, no. 4, pp. 503-513, 2007.

[40] T. Butsuen, "The Design of Semi-Active Suspensions for Automotive Vehicles," Ph.D. Dissertation, MIT, 1989.

[41] "http://www.moticont.com/lvcm-051-089-01.htm," Linear Voice Coil Motor Actuator LVCM-051-089-01, MotiCont: a Motion Company.

[42] "http://www.omega.ca/pptst_eng/LCCA.html," High Accuracy S-Beam Load Cells, Omega.

[43] "Data Sheet: Instrumentation Amplifier AD620," Analog Devices.

[44] "Data Sheet: Series RC13 - Potentiometric Linear Transducer," Megatron Elektronik GmbH & Co. KG, Munchen.

[45] "https://www.cytron.com.my/c-93-dc-motor-driver/p-md10c," Cytron 10Amp DC Motor Driver, Cytron Technologies Sdn. Bhd., Penang.

[46] "Data Sheet: TS-ADC16," Technologic Systems.

[47] "Data Sheet: TS-7300," Technologic Systems.

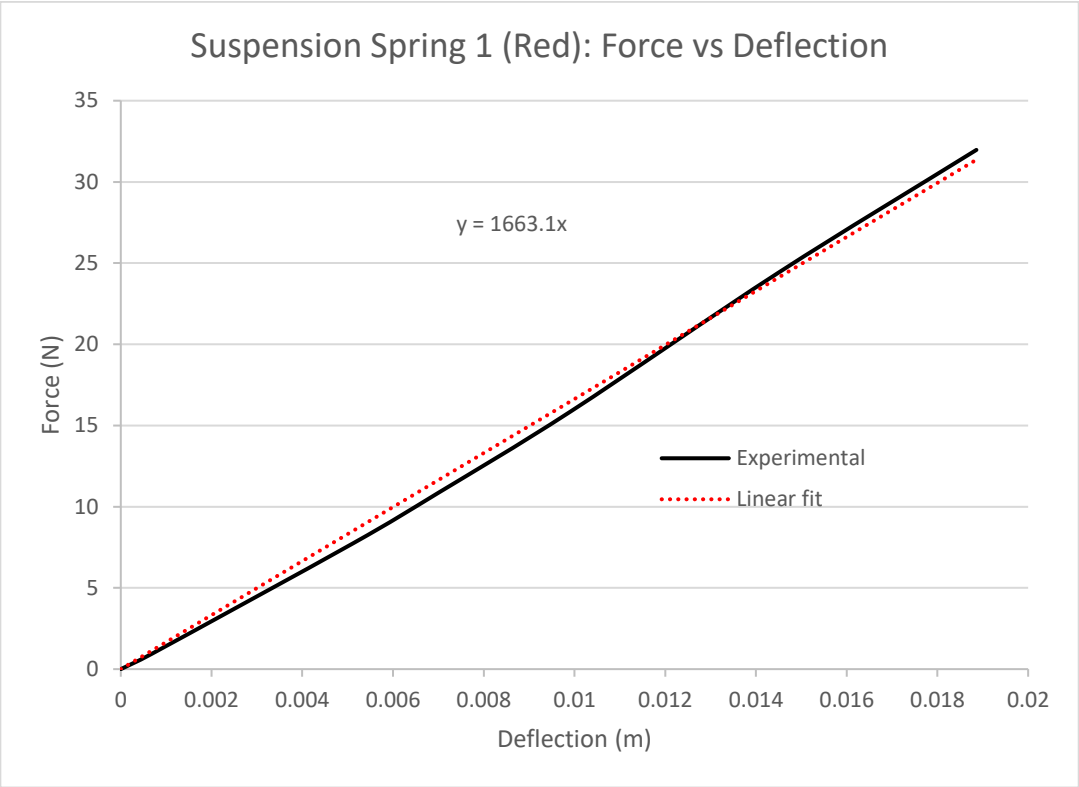
[48] "http://www.20sim4c.com/webhelp/index.html?fpga_dio2_default.htm," FPGA DIO2 Default, 20-sim 4C.

[49] "20-sim 4C 2.1 Reference Manual," Controllab Products B.V., Enschede, 2013.

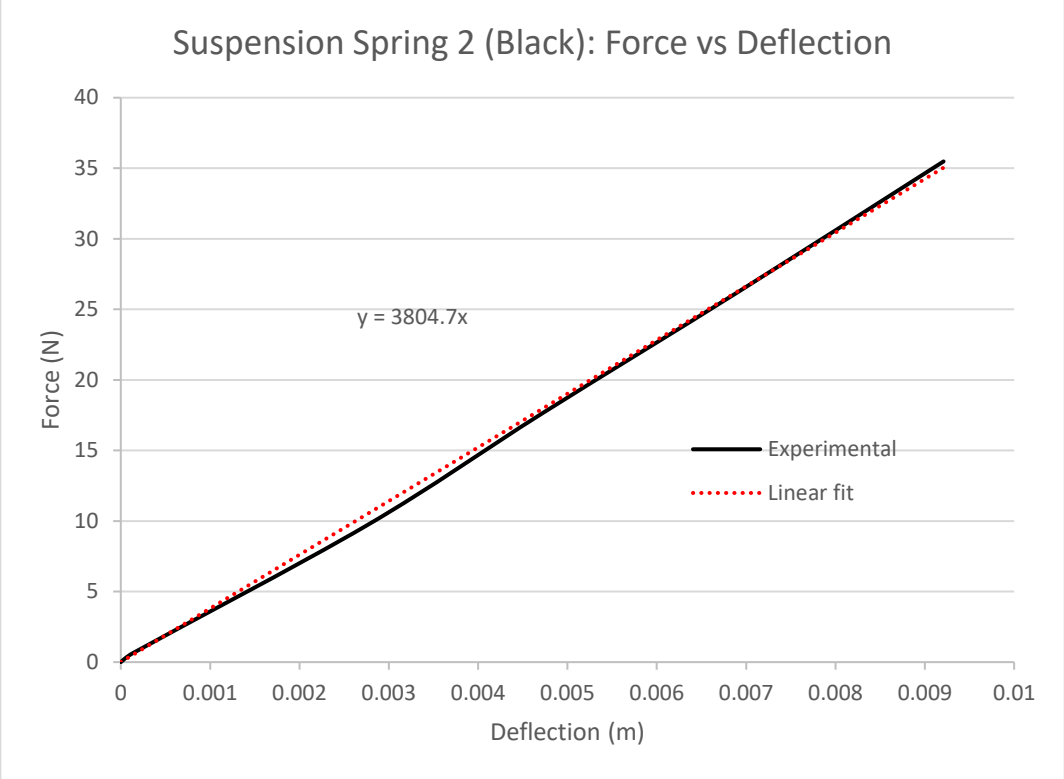
Appendix A

Spring Test

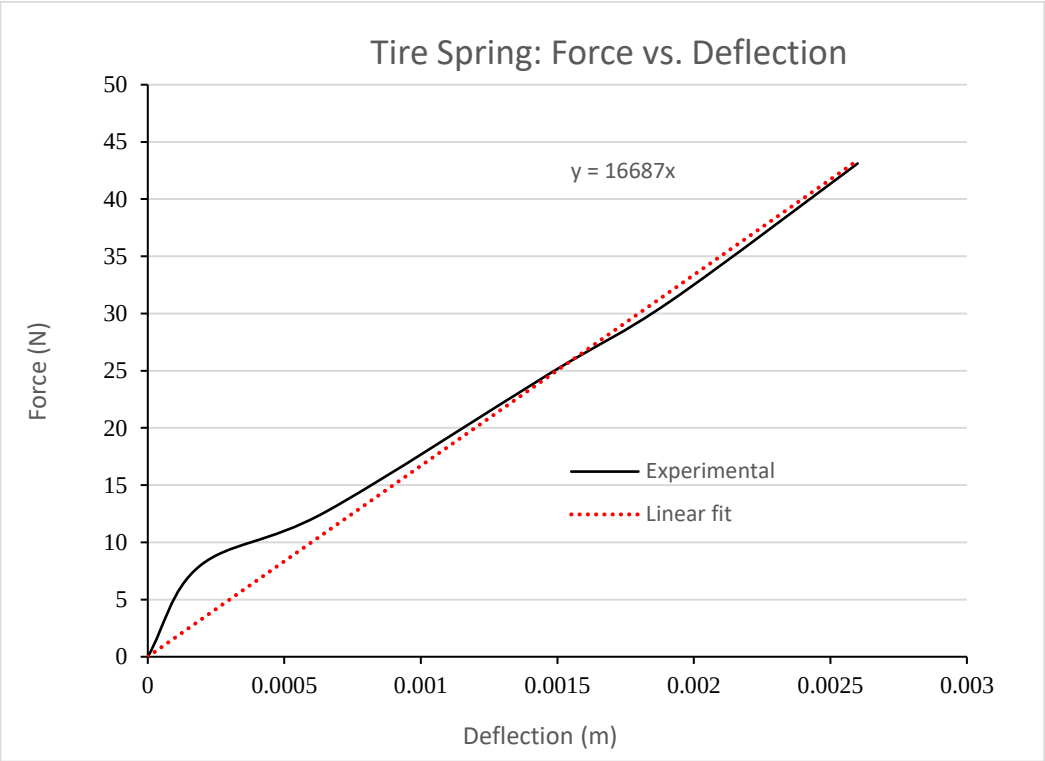
Suspension Spring 1 (Red)		
Mass (gm)	Deflection (mm)	Force (N)
0	0	0
97	0.69	0.95157
788	5.11	7.73028
1105	6.99	10.84005
1618	9.92	15.87258
2489	14.5	24.41709
3259	18.86	31.97079



Suspension Spring 2 (Black)		
Mass (gm)	Deflection (mm)	Force (N)
0	0	0
63	0.13	0.61803
990	2.76	9.7119
1798	4.72	17.63838
2678	6.92	26.27118
3617	9.21	35.48277



Tire Spring		
Mass (gm)	Deflection (mm)	Force (N)
0	0	0
808	0.19	7.92648
1289	0.65	12.64509
2520	1.47	24.7212
3161	1.91	31.00941
4395	2.6	43.11495


Tire Spring: Force vs. Deflection
y = 16687x
Experimental
Linear fit
Force (N)
Deflection (m)
0
5
10
15
20
25
30
35
40
45
50
0
0.0005
0.001
0.0015
0.002
0.0025
0.003

Appendix B

20-sim Codes

Sprung Mass (m_s):

```
parameters
        real i = 8.243; // including actuator body mass.
equations
   state = int(p.e);
   p.f = state / i;
mass = i;
ms = i;
```

Unsprung Mass (m_u):

```
parameters
        real i = 2.5; // including actuator coil mass.
equations
   state = int(p.e);
```

```
   p.f = state / i;
mass = i;
```

Rail Friction for Sprung Mass Motion (R_s):

```
parameters
        real r1 = 8;
        real r2 = 10;
        real r3 = 1.25;
        real fmax = 1.0; // max. absolute flow for Coulomb friction region
variables
        real f; // flow
        real farg;
equations
        f = p.f;

if abs(f) > fmax then
        farg = fmax*sign(f);
else
        farg = f;
end;

        p.e = r1*(2/(1+exp(-r2*farg))-1)+r3*f^2*sign(f); // Coulomb & Drag friction
```

Rail Friction for Unsprung Mass Motion (R_u):

```
parameters
        real r1 = 5;
        real r2 = 10;
        real r3 = 1.25;
        real fmax = 0.9; // max. absolute flow for Coulomb friction region
variables
        real f; // flow
        real farg;
equations
        f = p.f;


if abs(f) > fmax then
        farg = fmax*sign(f);
else
        farg = f;
end;


        p.e = r1*(2/(1+exp(-r2*farg))-1)+r3*f^2*sign(f); // Coulomb & Drag friction
```

Suspension Damping (b_{susp}):

```
parameters
        real r1 = 10.5;
        real r2 = 100;
```

```
        real r3 = 25;
        real fmax = 0.11; // max. absolute flow for Coulomb friction region
variables
        real f; // flow
        real farg;
equations
        f = p.f;

if abs(f) > fmax then
        farg = fmax*sign(f);
else
        farg = f;
end;

        p.e = r1*(2/(1+exp(-r2*farg))-1)+r3*f; // Coulomb & Viscous friction
```

Suspension Spring (k_{susp}):

```
parameters
        real c = 8.64e-4;
equations
        state = int(p.f);
        p.e = state / c;
```

Tire Spring (k_t):

```
parameters
        real c = 5.99e-005;
equations
        state = int(p.f);
        if state > 0 then
                p.e = state / c;
        else
                p.e = 0;
                end;
```

Road Input (v_r):

```
parameters
//      real amplitude = 0.05;                  // trial & error
//      real omega = 3.1416 {rad/s};

        real amplitude = 0.1257;                // 2mm 10Hz
        real omega = 62.83 {rad/s};

//      real amplitude = 0.1885;                // 2mm 15Hz
//      real omega = 94.25 {rad/s};
//      real amplitude = 0.016;                 // 0.165mm 15Hz MTS
//      real omega = 94.25 {rad/s};
```

```
//      real amplitude = 0.031;                 // 10mm 0.5Hz
//      real omega = 3.1416 {rad/s};

//      real amplitude = 0.063;                 // 10mm 1.0Hz
//      real omega = 6.2832 {rad/s};

//      real amplitude = 0;                 // 0mm 1Hz
//      real omega = 6.2831 {rad/s};
//      real amplitude = 0.047;                 // 7.5mm 1Hz MTS
//      real omega = 6.2831 {rad/s};

variables
        boolean hidden change;
        real hidden half;
equations
        "calculate at least 2 points per cycle to get a triangle"
        half = pi / omega;
        change = frequencyevent (half, half / 2);

        "calculate the sine wave"
        output = amplitude * sin (omega * time);
```

Actuator (MSe):

```
parameters
//      real G[1,4] = [-1152, -496, 5, 499.5];  //trial & error
```

```
//      real G[1,4] = [-500, -800, -1000, 0];  //trial & error Rideout _ Mehedi road
//      real G[1,4] = [-50, -500, -10, 0];  //trial & error Rideout _ Rideout road

// updated
        real G[1,4] = [-1152, -21, 5, 24.5];
//heavily weighted ride quality using A in Riccati
//      real G[1,4] = [-992.39, 27.42, 8.77, 0.51];
//moderately weighted ride quality using A in Riccati
//      real G[1,4] = [-332.95, 109.23, -597.28, -52.53];
//heavily weighted tire/susp deflection using A in Riccati

// 1st attempt
//      real G[1,4] = [-1150, -25.5, 1.1, 31.1];
//heavily weighted ride quality using A in Riccati
//      real G[1,4] = [-1015.5, 19.2, -70.7, 5.6];
//moderately weighted ride quality using A in Riccati
//      real G[1,4] = [-448.45, 90.5567, -709.9935, -40.2846];
//heavily weighted tire/susp deflection using A in Riccati

variables
        real flow;
        real F[1,4]; // individual actuator force component
equations
        p.e = -G*x;
        flow = p.f;
```

```
F = -G .* transpose(x);
```

Actuator Voltage (MSe):

```
variables
    real flow;
equations
    p.e = effort;
    flow = p.f;
```

Actuator Resistance (Rcoil):

```
parameters
    real r = 6.0; //Actuator spec
equations
    p.e = r * p.f;
```

Gyrator (GY):

```
parameters
    real r = 10.1; //Actuator spec
equations
    p1.e = r * p2.f;
    p2.e = r * p1.f;
```

Spring (k_inv):

```
parameters
    real c = 0.000601286; //Spring 1 (red), k = 1663.1 N/m
```

```
equations
        state = int(p.f);
        p.e = state / c;
```

Unsprung Mass Plus Coil Mass (musPlusmcoil):

```
parameters
        real i = 2.5; //m_us + m_coil
equations
   state = int(p.e);
   p.f = state / i;
```

Rail Friction (railFriction):

```
parameters
        real r1 = 10;
        real r2 = 10;
        real r3 = 1.25;
        real fmax = 0.9; // max. absolute flow for coulomb friction region
variables
        real f; // flow
        real farg;
equations
        f = p.f;

if abs(f) > fmax then
        farg = fmax*sign(f);
```

```
else
        farg = f;
end;

        p.e = r1*(2/(1+exp(-r2*farg))-1)+r3*f^2*sign(f); // Coulomb & Drag friction
```

mToVconversion:

```
parameters
        real m = 7.93;                  // slope
        real c = 0.02;  // y intercept
        // L(mm) = m * V - c
equations
        output = ((input * 1000) + c) / m; // meter to V conversion
```

UnitSignalToVconversion:

```
parameters
        real c = 30;            // limiting V
equations
        output = input * c; // V to unit signal conversion
```

SignalMonitor_TopLPDT:

```
variables
        real interesting plot;
        real initVal;
equations
```

```
        if time <= 0.002 then
                initVal = (7.93*input - 0.02) / 1000; // Obtaining initial value of LPDT
        end;
        plot = (7.93*input - 0.02) / 1000; // V to m conversion
        output = plot - initVal; // zero bias
```

RefDistance:

```
parameters
        real amplitude = 0.01;                  // amplitude of the wave
        real omega = 5.0 {rev/s};               // frequency of the wave
variables
        boolean hidden change;
        real hidden half;
        real T; // period
equations
        "calculate at least 2 points per cycle to get a triangle"
        half = pi / omega;
        change = frequencyevent (half, half / 2);

        T = 2 * pi / omega; // period

        "calculate the sine wave"
        output = amplitude * sin ( omega * time); // symmetric
//      output = amplitude * sin ( omega * time) + 0.005; // asymmetric
```

```
//      if time < (T/2) then
//              output = 0; // initial zero output for half period
//      end;
```

ControlVoltage:

```
parameters
        real maximum = 30;
        real minimum = -30;
equations
        output = limit (input, minimum, maximum);
```

VtoUnitSignalconversion:

```
parameters
        real c = 30;            // limiting V
equations
        output = input / c; // V to unit signal conversion
```

FtoVconversion:

```
parameters
        real m = 36.696;        // slope (N/V)
        real c = 11.171; // y intercept (N)
        // LC horizontal
equations
        output = (-input - c) / m; // F to V conversion
//      compression: +ve from spring
```

```
//      compression: -ve voltage from LC
```

iToVconversion:

```
parameters
        real c = 0.1;    // factor = 100mV/A
equations
        output = c * input; // i to V conversion
```

SignalMonitor_LC:

```
parameters
        real m = 36.696;       // slope (N/V)
        real c = 11.171; // y intercept (N)
        // LC horizontal
variables
        real interesting plot;
        real initVal;
equations
        if time <= 0.002 then
                initVal = input * m + c; // Obtaining initial value of LC
        end;
        plot = input * m + c; // V to F conversion
        output = plot; //- initVal; // zero bias
```

RefForce:

```
parameters
        real amplitude = 20;                // amplitude of the force (N)
        real omega = 0.5 {rev/s};           // angular frequency of the wave
variables
        boolean hidden change;
        real hidden half;
equations
        "calculate at least 2 points per cycle to get a triangle"
        half = pi / omega;
        change = frequencyevent (half, half / 2);

        "calculate the sine wave"
        output = amplitude * sin ( omega * time);
```

SignalMonitor_CurrentProbe:

```
variables
        real interesting plot;
        real initVal;
        real output;
equations
        if time <= 0.002 then
                initVal = input / 0.1; // Obtaining initial value of Current Probe
        end;
        plot = input / 0.1; // V to i conversion, 100mV/A
```

```
        output = plot - initVal; // zero bias
```

mcoil:

```
parameters
        real i = 0.195; // mass of actuator coil 0.195 kg
equations
   state = int(p.e);
   p.f = state / i;
mass = i;
```

Wcoil:

```
variables
        real flow;
equations
        p.e = -mass*9.81;
        flow = p.f;
```

kLC:

```
parameters
        real c = 1e-005; // load cell stiffness 100000 N/m (very high stiffness)
equations
        state = int(p.f);
        p.e = state / c;
```

StateMatrix:

equations

output = [input1; input2; input3; input4];

RefForce (Realistic Model):

parameters

// real G[1,4] = [-1152, -496, 5, 499.5]; //trial & error

// real G[1,4] = [-500, -800, -1000, 0]; //trial & error Rideout _ Mehedi road

// real G[1,4] = [-50, -500, -10, 0]; //trial & error Rideout _ Rideout road

// updated

real G[1,4] = [-1152, -21, 5, 24.5]; //heavily weighted ride quality using A in Riccati

// real G[1,4] = [-992.39, 27.42, 8.77, 0.51]; //moderately weighted ride quality using A in Riccati

// real G[1,4] = [-332.95, 109.23, -597.28, -52.53]; //heavily weighted tire/susp deflection using A in Riccati

// 1st attempt

// real G[1,4] = [-1150, -25.5, 1.1, 31.1]; //heavily weighted ride quality using A in Riccati

// real G[1,4] = [-1015.5, 19.2, -70.7, 5.6]; //moderately weighted ride quality using A in Riccati

// real G[1,4] = [-448.45, 90.5567, -709.9935, -40.2846]; //heavily weighted tire/susp deflection using A in Riccati

variables

```
        real F[1,4]; // individual actuator force component
equations
        F = -G .* transpose(x);
        RefForce = -G*x;
```

Flc_initial:

```
parameters
        real C = 1.913;                 // LC initial
equations
        output = C;
```

Appendix C

Matlab Codes

Modal Analysis:

```
clear all;
close all;

% Assumption:
% the system is undamped and
% no force is applied

ms = 7.088 % sprung mass
mu = 2.305 % unsprung mass

ks = 1157.25 % suspension stiffness
kt = 16687 % tire stiffness

M = [ms 0; 0 mu] % system mass matrix
K = [ks -ks; -ks ks+kt] % stiffness matrix

[v D] = eig(M\K) % eigen values
lambda = [v D] % eigen vector

w = sqrt(D) % natural frequency matrix
```

```
w1 = w(1,1) % 1st natural frequency
w2 = w(2,2) % 2nd natural frequency

syms p11 p12 p21 p22;
p = [p11 p12; p21 p22] % mass-normalized modal
matrix (symbolic)
B = p'*M*p == [1 0; 0 1] % condition of mass
normalization

% solving equations for modal matrix
Psol = solve([B(1,1), B(1,2), B(2,1), B(2,2)],
[p11 p12 p21 p22])

% mass-normalized modal matrix (solved)
P = [Psol.p12 Psol.p11; Psol.p22 Psol.p21]

% checking the validity of the solution
P'*M*P % checking mass normalization
sqrt(P'*K*P) % checking the natural frequencies
```

LQR Controller Design:

```
clear all;
close all;

% Parameters
ms = 8.243; % including actuator body mass
mu = 2.5; % including actuator coil mass
bs = 25; % linear assumption
bt = 0;
b1 = 10; % linear assumption
b2 = 10; % linear assumption
ks = 1157.25;
kt = 16687;

% Cost Function
```

```
% zsddot^2 + p1(susp.def.)^2 + p2(zsdot)^2 +
p3(tire def.^2) + p4(zudot)^2


% Weighting factors - heavily weighted ride
quality
p1=0.4;
p2=0.16;
p3=0.4;
p4=0.16;

% Moderately weighted ride quality
% p1=400;
% p2=16;
% p3=400;
% p4=16;

% Heavily weighted tire/susp deflection
% p1=10000;
% p2=100;
% p3=100000;
% p4=100;


% State variables
% x1 = zs-zu
% x2 = zsdot
% x3 = zu-zr
% x4 = zudot

% Matrix
% xdot = Ax + BFa + Lzrdot
%
A = [0 1 0 -1;
    -ks/ms -(bs+b1)/ms 0 bs/ms;
    0 0 0 1;
    ks/mu bs/mu -kt/mu -(bs+bt+b2)/mu];
```

```
B = [0;
     1/ms;
     0;
     -1/mu];

% gravity effect
% D = [0;
%      1;
%      0;
%      1];

L = [0;
     0;
     -1;
     0];

Q = [ks^2/ms^2+p1 bs*ks/ms^2 0 -bs*ks/ms^2;
     bs*ks/ms^2 bs^2/ms^2+p2 0 -bs^2/ms^2;
     0 0 p3 0;
     -bs*ks/ms^2 -bs^2/ms^2 0 bs^2/ms^2+p4];

N = [-ks/ms^2;
     -bs/ms^2;
     0;
     bs/ms^2];

R = 1/ms^2;

S = N;

% Rajamani Eq. 11.16
% Riccati Eq
[P, Lam, G1] = care(A, B, Q, R, S);
% P = unknown
% L = closed loop eigenvalues
% G = gain matrix

% Rajamani Eq. 11.17
```

```
G1
G2 = inv(R)*(B'*P+N')
% Fa = -Gx
```

FFT Analysis (Load Cell Voltage):

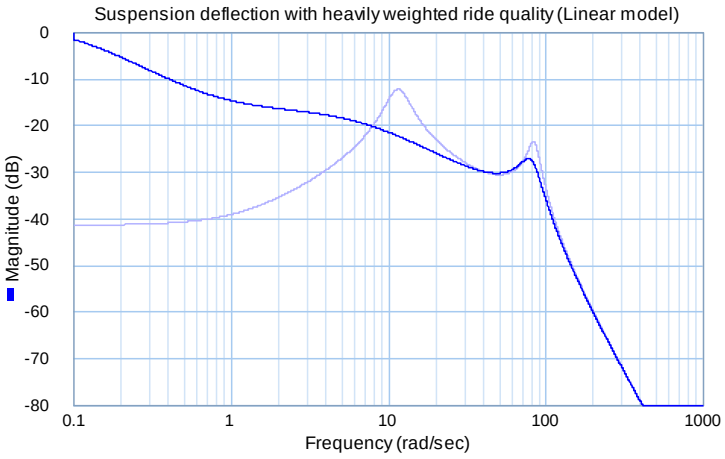
```
clear all
close all

% RefSignal = RefForce = 40 @ 0.5 Hz

% load data file
load LCvolt.txt

% unfiltered LCvoltage data: column 15th
LCvolt = LCvolt(:, 15);

f_Samp = 180; % approximately
f_Nyquist = f_Samp/2;

Yraw = fft(LCvolt); % FFT values, each
multiplied by N (total number of
% data pt.)
SY = size(Yraw); % gives number of rows and
columns in Yraw
N = SY(1); % integer number of data values in
the raw spectrum data

% first N/2 data are for frequencies from 0 to
f_Nyquist
freqstep = f_Nyquist/(N/2); % frequency step per
data

% x values (frequencies)
for i = 1:N/2
    f(i) = (i-1)*freqstep;
end
```

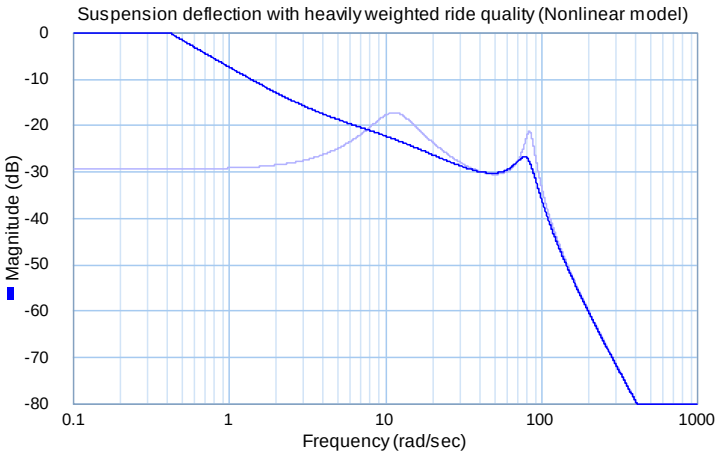
```
% Oh by the way, fft returns series elements
that are multiplied by
% N (the number of data points).
% Divide elements by N.
%
Y = Yraw/N;

% The last N/2 elements are complex conjugates
of the first N/2, except
% for the first one, which is for the real
component.
% Compute magnitudes of the first N/2 elements,
to turn them from
% imaginary to real.
%
for i = 1:N/2
    Ymag(i) = abs(Y(i));
end

% Plot magnitudes vs. frequency.  Plot the
"single-sided amplitude
% spectrum" by multiplying the magnitudes by 2.
This is because
% there are two coefficients for each frequency:
one in the first
% half of the vector of Fourier coefficients,
and one in the second
% (complex conjugate) half.
%
figure(1)
plot(f,2*Ymag);
% xlim([0,10]);
title('Unfiltered LCvolt, f_s = 180')
xlabel('Frequency [Hz]')
ylabel('Amplitude')
```

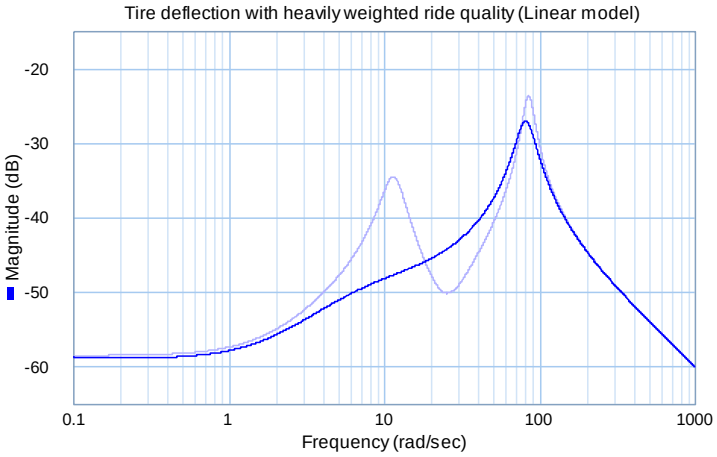
Appendix D

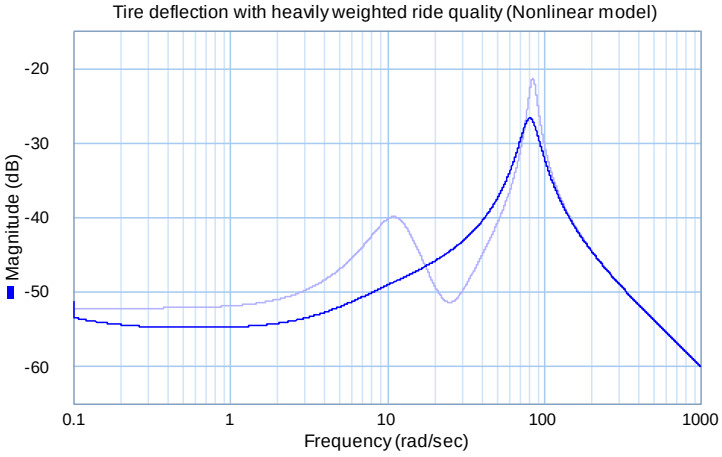
LQR Controller Performance Plots

Controller Performance – Linear Model vs. Nonlinear Model:

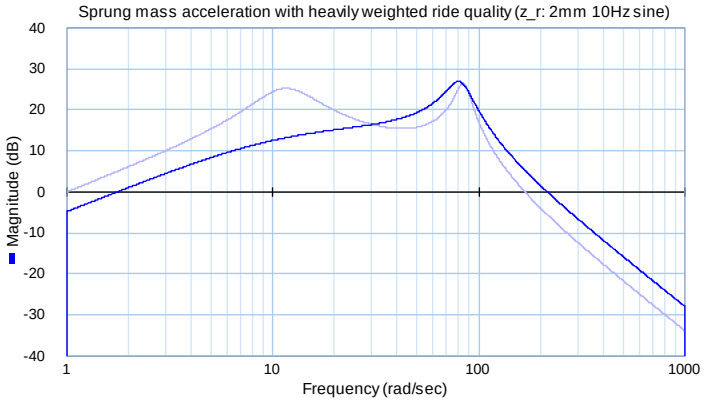


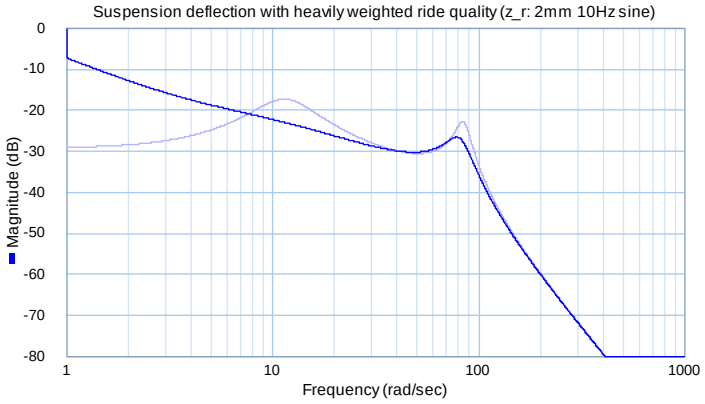

Suspension deflection with heavily weighted ride quality (Nonlinear model)
0
-10
-20
-30
-40
-50
-60
-70
-80
Magnitude (dB)
0.1
1
10
100
1000
Frequency (rad/sec)

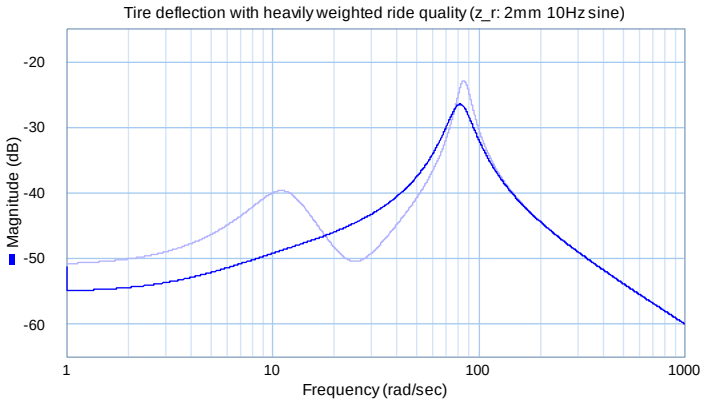

Tire deflection with heavily weighted ride quality (Linear model)
-20
-30
-40
-50
-60
Magnitude (dB)
0.1
1
10
100
1000
Frequency (rad/sec)

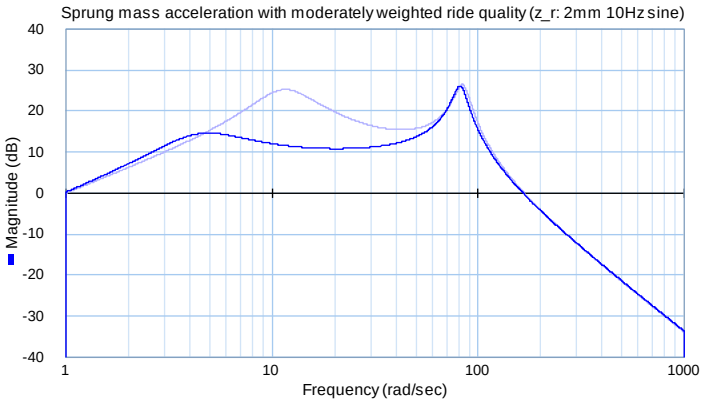

Tire deflection with heavily weighted ride quality (Nonlinear model)
-20
-30
-40
-50
-60
Magnitude (dB)
0.1
1
10
100
1000
Frequency (rad/sec)

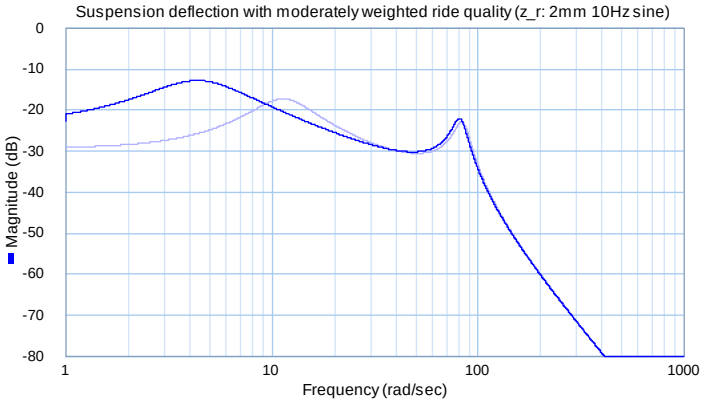
z_r = 2mm 10Hz sine:

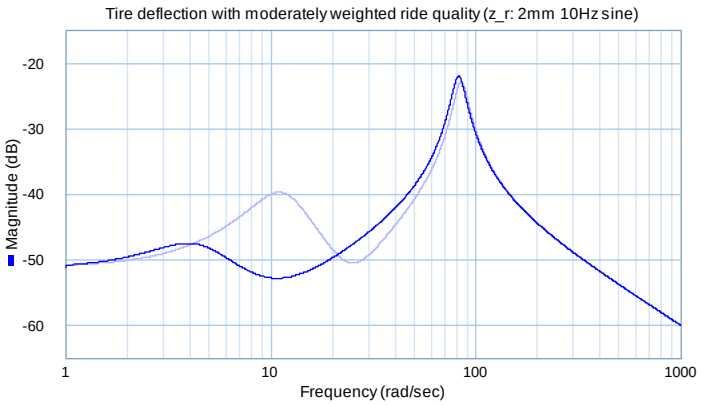




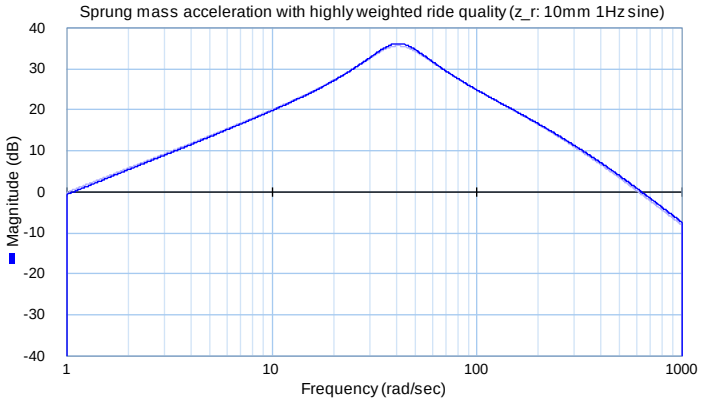


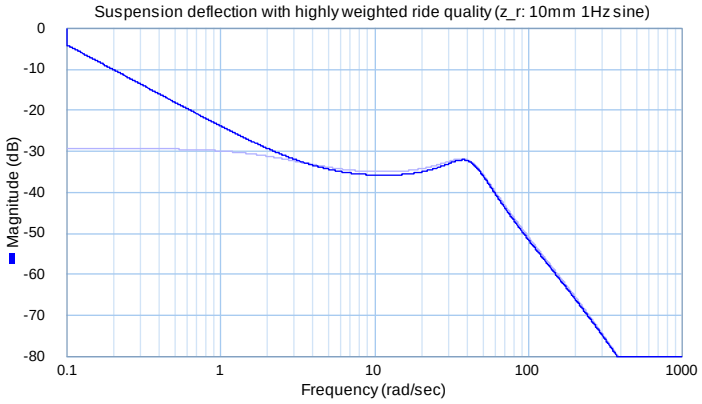

Sprung mass acceleration with moderately weighted ride quality (z_r: 2mm 10Hz sine)
Magnitude (dB)
40
30
20
10
0
-10
-20
-30
-40
1
10
100
1000
Frequency (rad/sec)

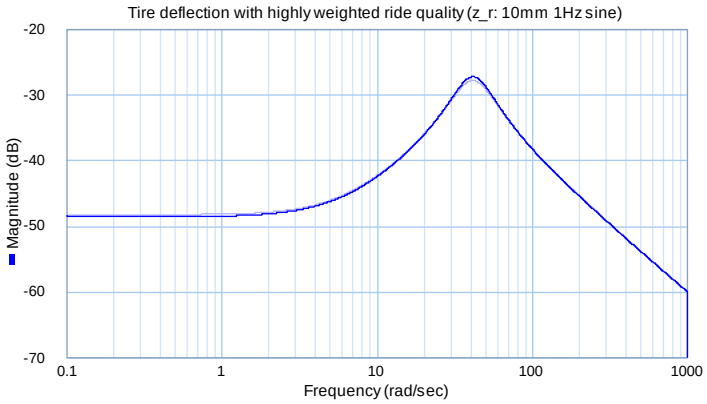

Suspension deflection with moderately weighted ride quality (z_r: 2mm 10Hz sine)
Magnitude (dB)
0
-10
-20
-30
-40
-50
-60
-70
-80
1
10
100
1000
Frequency (rad/sec)

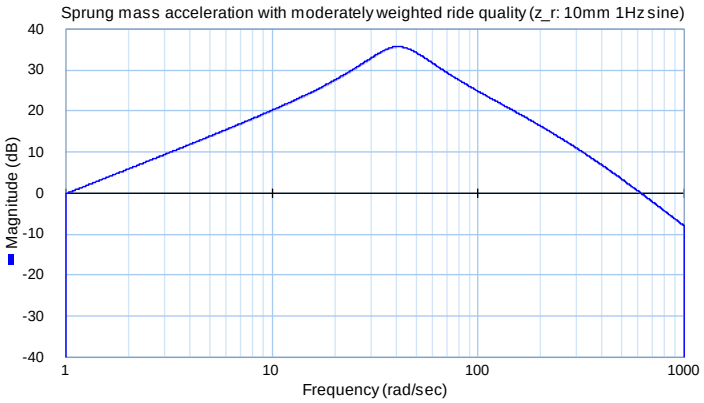

Tire deflection with moderately weighted ride quality (z_r: 2mm 10Hz sine)
Magnitude (dB)
-20
-30
-40
-50
-60
1
10
100
1000
Frequency (rad/sec)

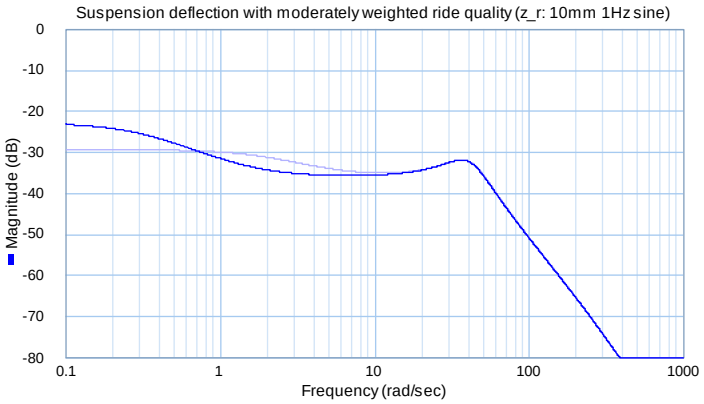
z_r = 10mm 1Hz sine:

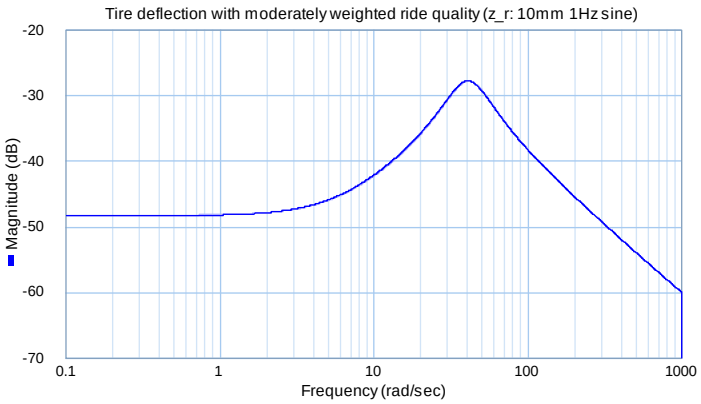




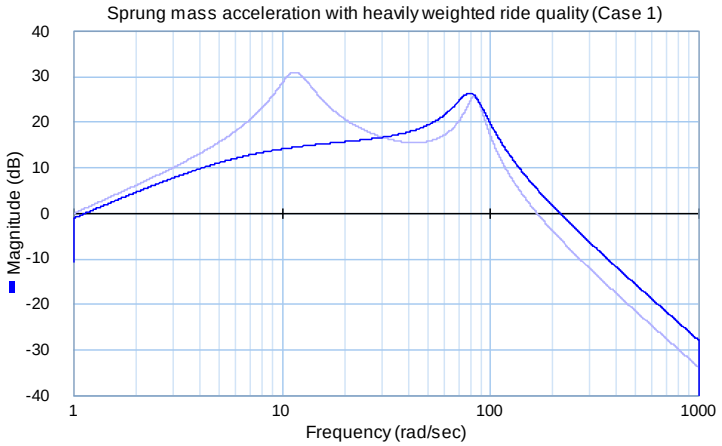


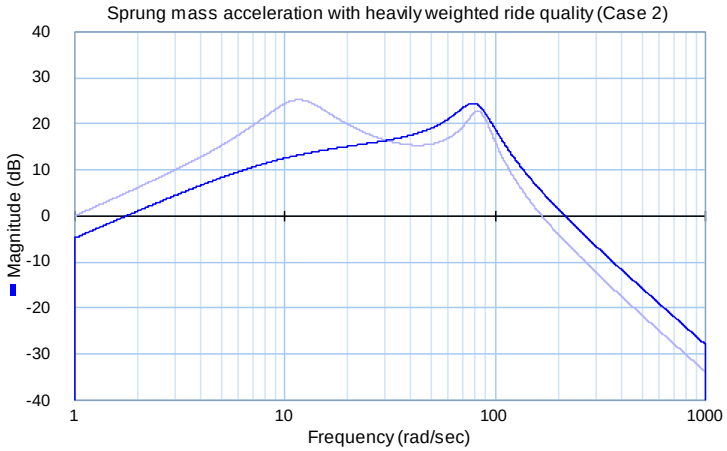

Sprung mass acceleration with moderately weighted ride quality (z_r: 10mm 1Hz sine)
Magnitude (dB)
40
30
20
10
0
-10
-20
-30
-40
1
10
100
1000
Frequency (rad/sec)

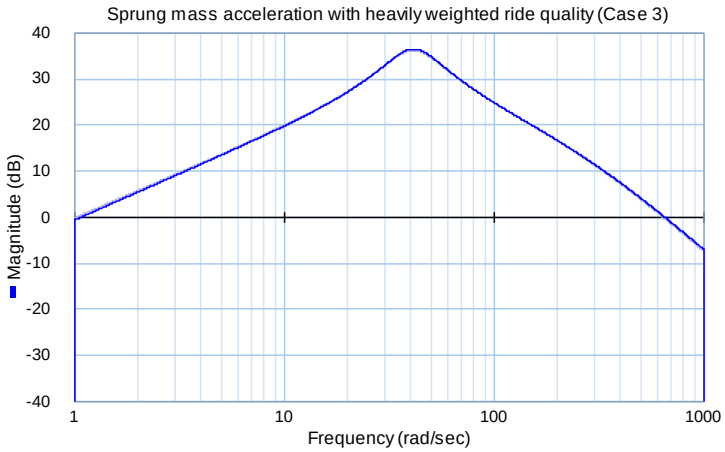

Suspension deflection with moderately weighted ride quality (z_r: 10mm 1Hz sine)
Magnitude (dB)
0
-10
-20
-30
-40
-50
-60
-70
-80
0.1
1
10
100
1000
Frequency (rad/sec)

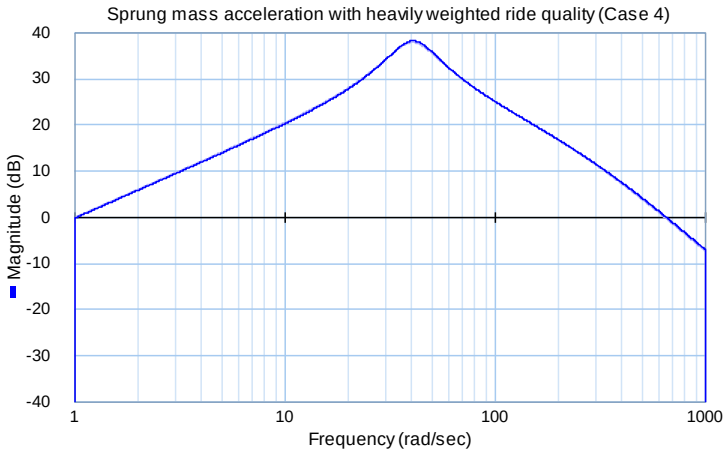

Tire deflection with moderately weighted ride quality (z_r: 10mm 1Hz sine)
Magnitude (dB)
-20
-30
-40
-50
-60
-70
0.1
1
10
100
1000
Frequency (rad/sec)

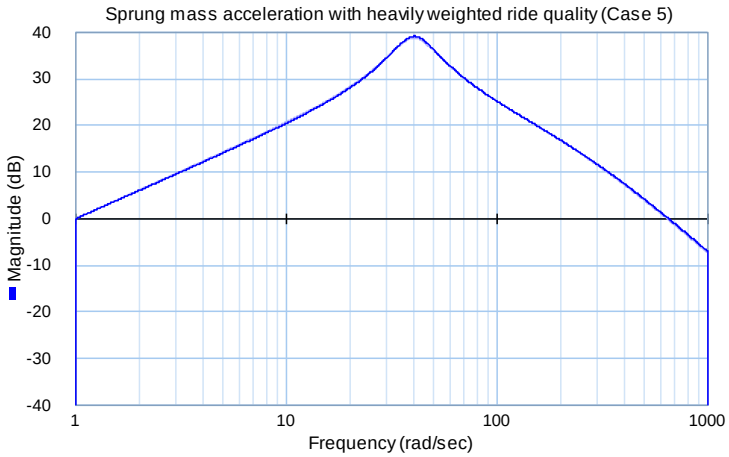
One-peak Bode Plot Investigation:



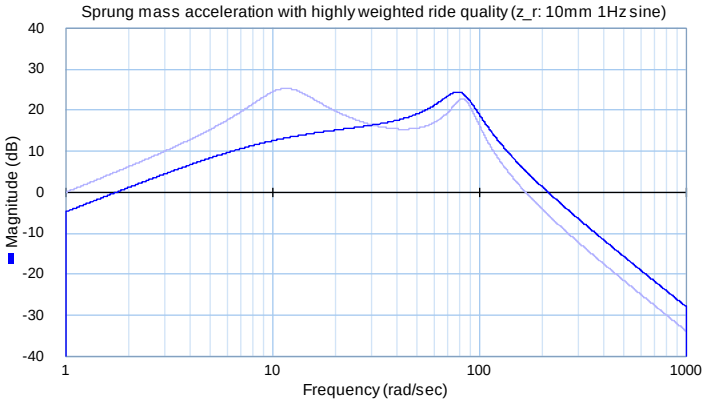


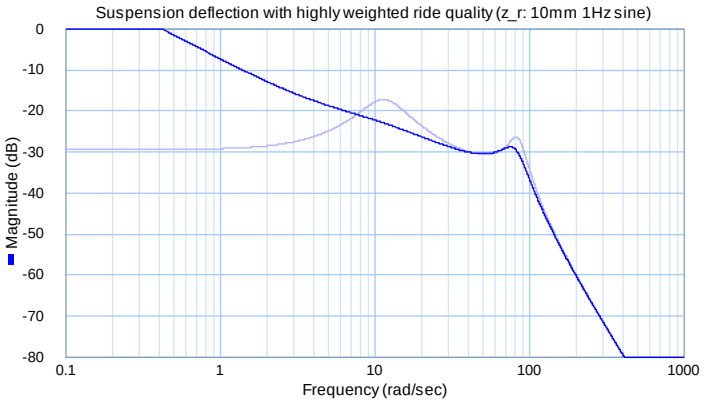


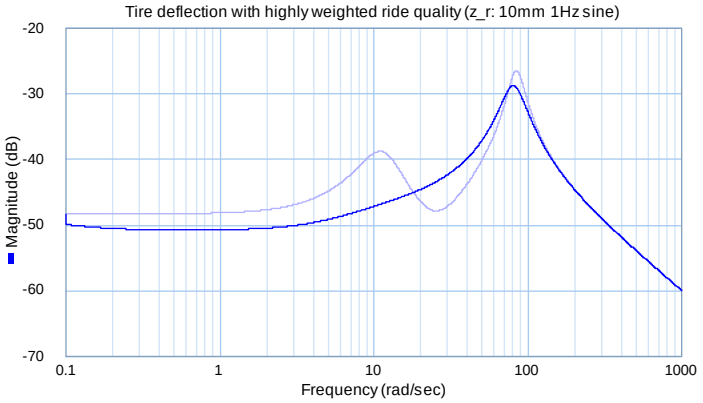


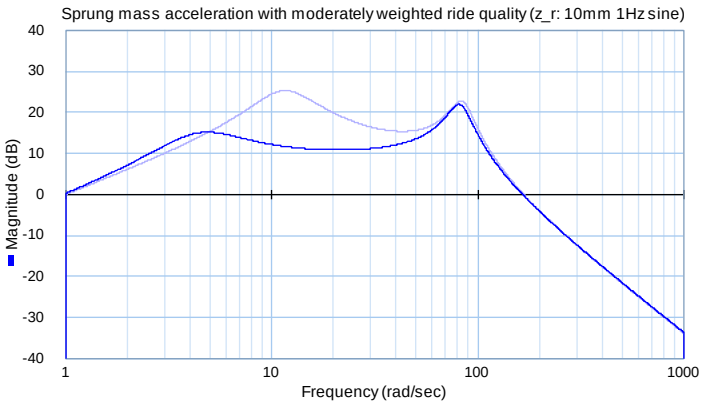


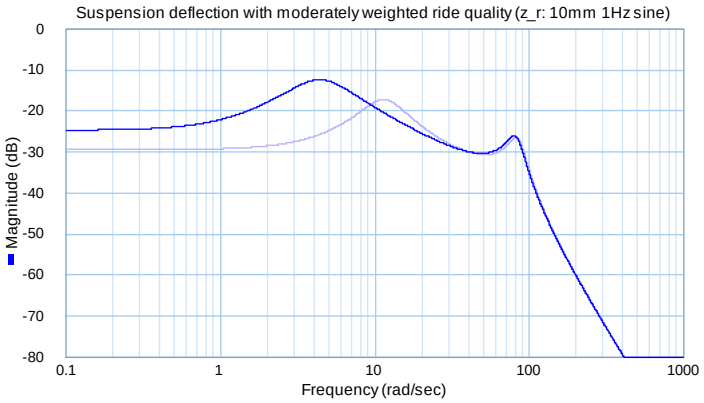
z_r = 10mm 1Hz sine (Linear Suspension Damping = 25 Ns/m):

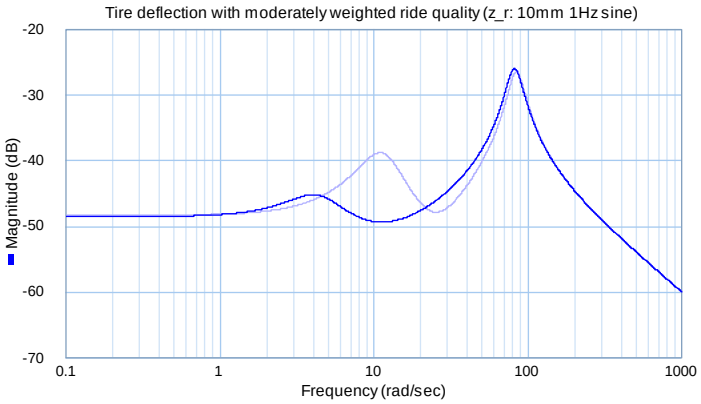



Suspension deflection with highly weighted ride quality (z_r: 10mm 1Hz sine)
Magnitude (dB)
0
-10
-20
-30
-40
-50
-60
-70
-80
0.1
1
10
100
1000
Frequency (rad/sec)

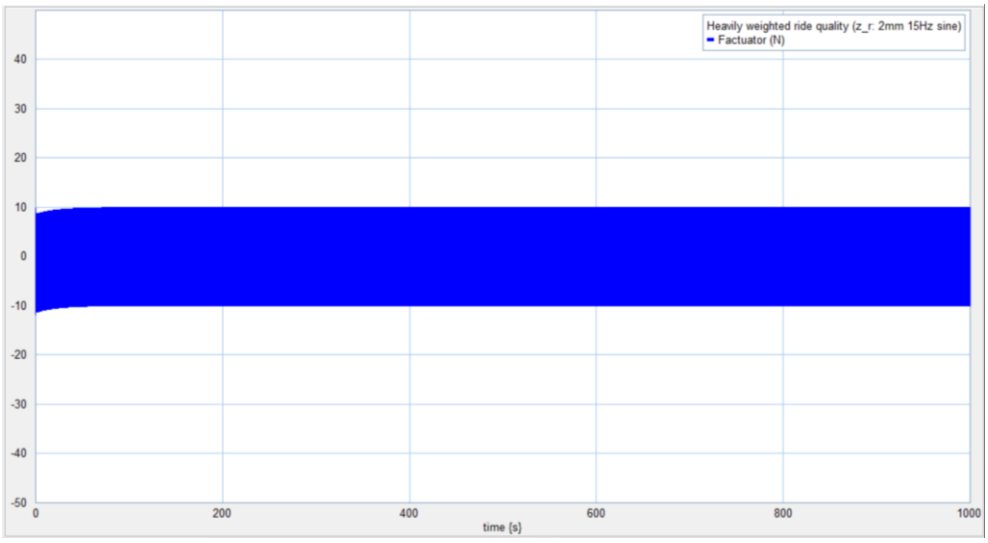

Tire deflection with highly weighted ride quality (z_r: 10mm 1Hz sine)
Magnitude (dB)
-20
-30
-40
-50
-60
-70
0.1
1
10
100
1000
Frequency (rad/sec)

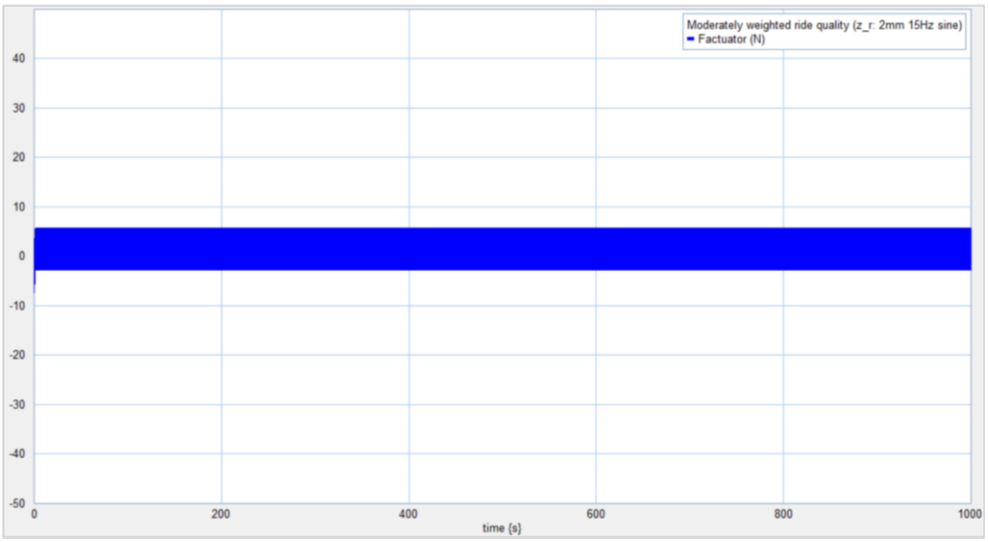

Sprung mass acceleration with moderately weighted ride quality (z_r: 10mm 1Hz sine)
Magnitude (dB)
40
30
20
10
0
-10
-20
-30
-40
1
10
100
1000
Frequency (rad/sec)

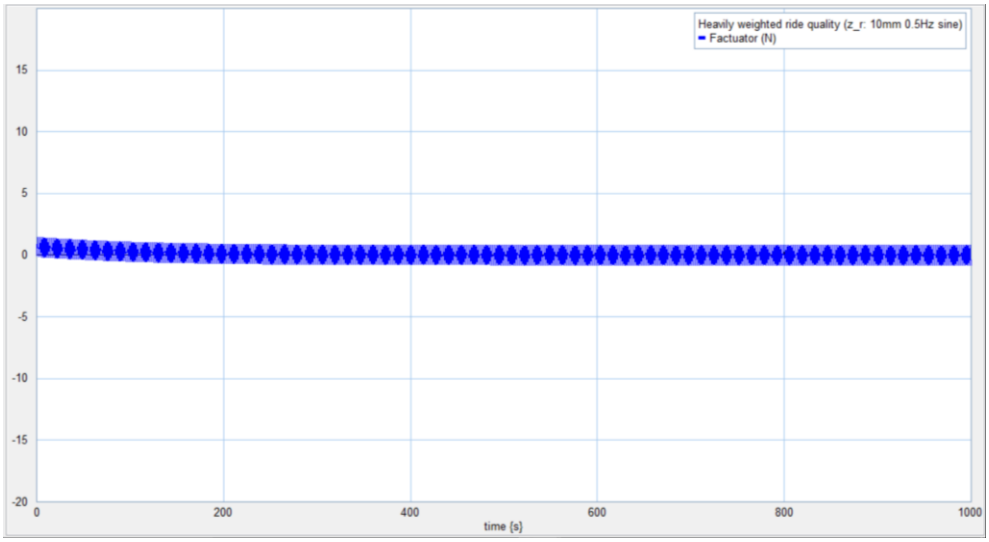


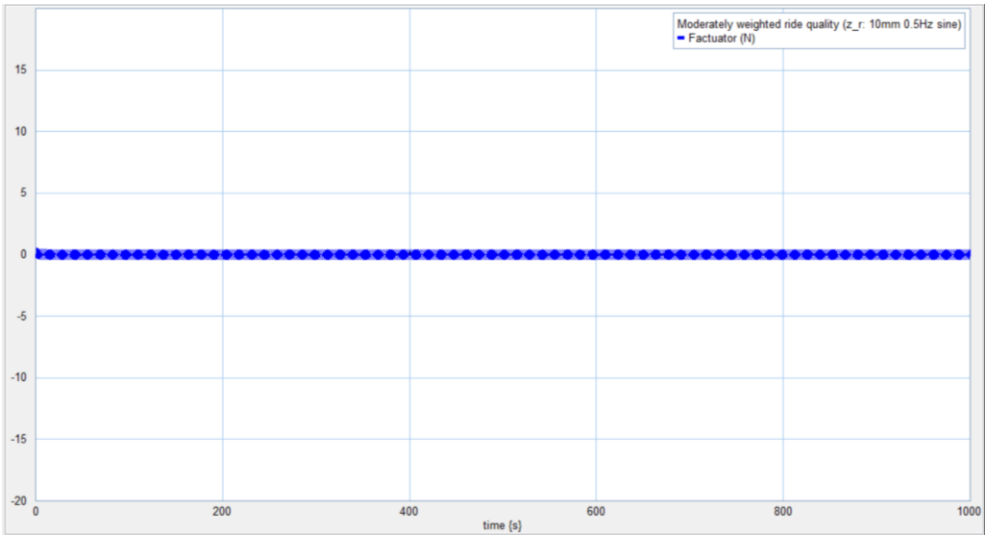


Actuator Force for Sinusoidal Road Inputs:




Moderately weighted ride quality (z_r: 2mm 15Hz sine)
Factuator (N)
40
30
20
10
0
-10
-20
-30
-40
-50
0
200
400
600
800
1000
time (s)

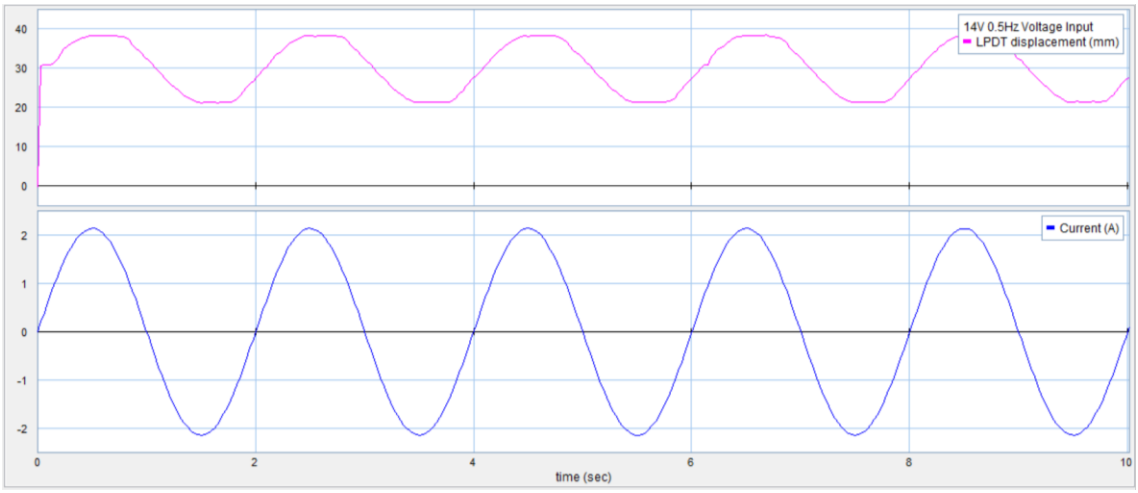

Heavily weighted ride quality (z_r: 10mm 0.5Hz sine)
Factuator (N)
15
10
5
0
-5
-10
-15
-20
0
200
400
600
800
1000
time (s)

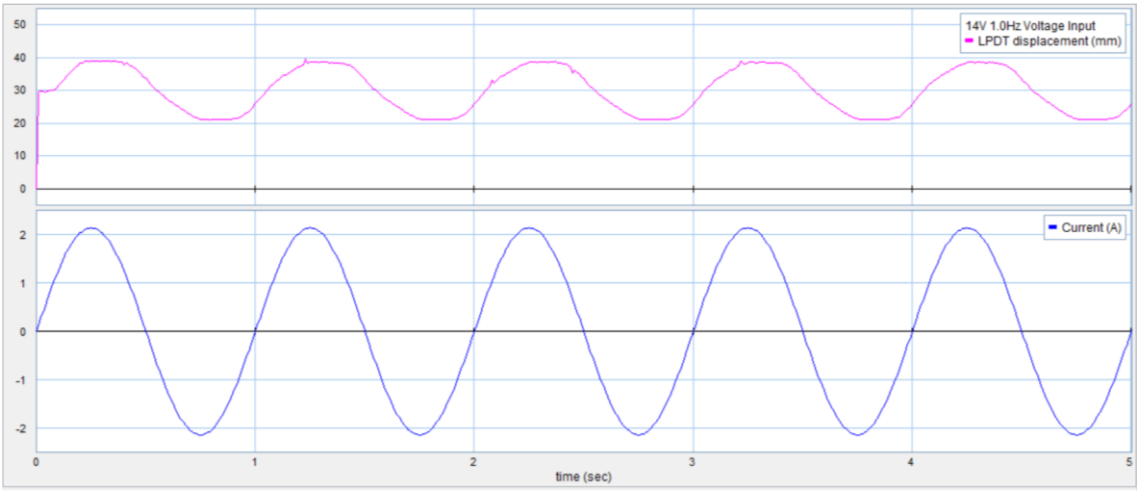

Moderately weighted ride quality (z_r: 10mm 0.5Hz sine)
Factuator (N)
15
10
5
0
-5
-10
-15
-20
0
200
400
600
800
1000
time (s)

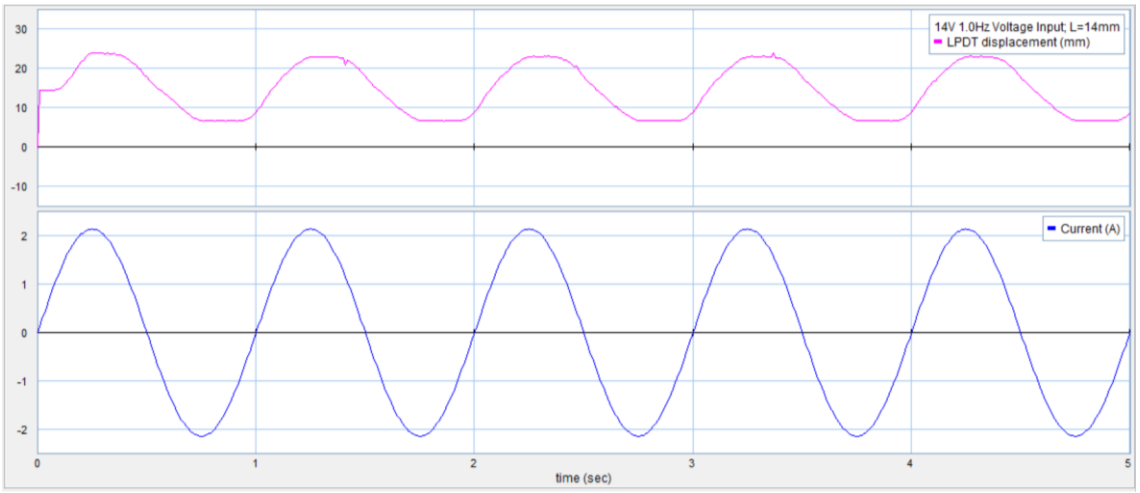
Appendix E

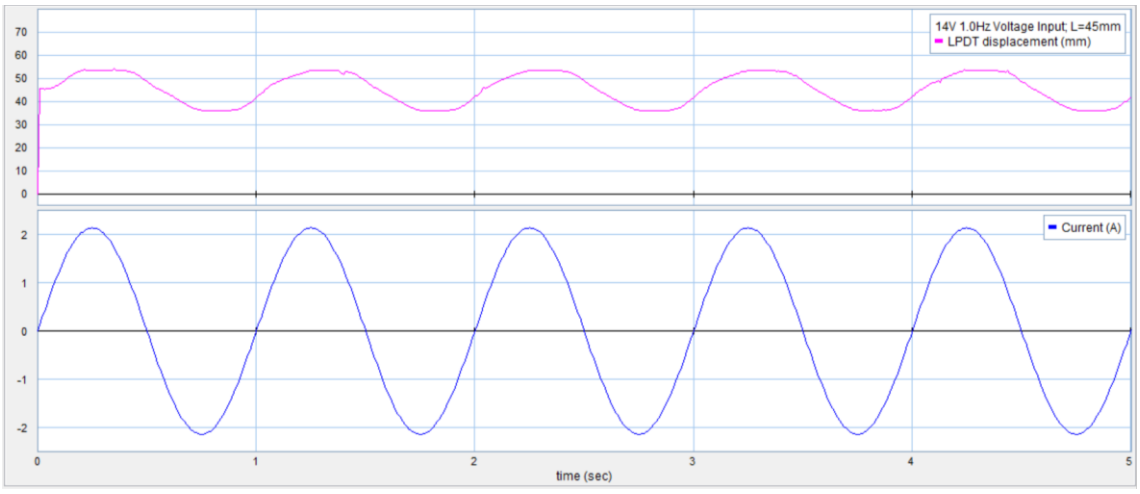
Actuator Test

Open Loop Test:

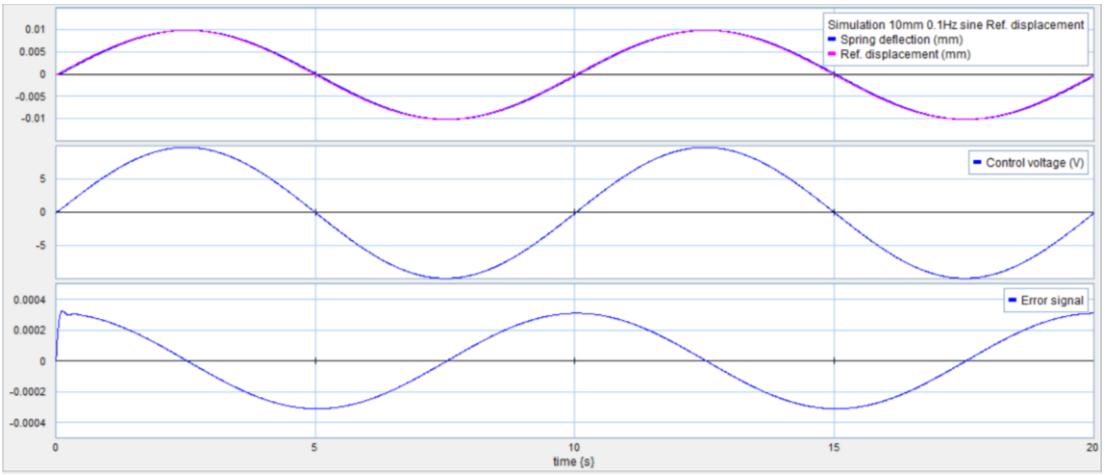


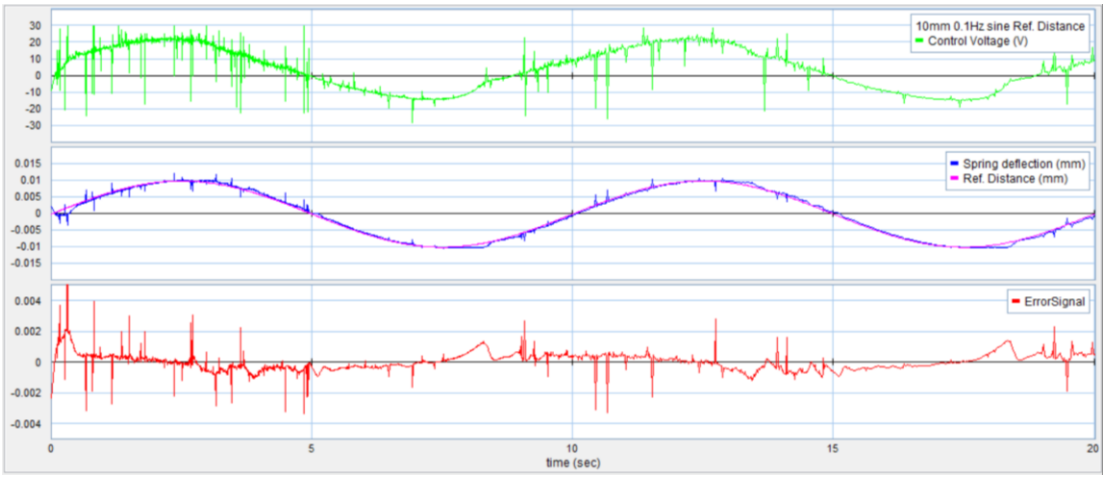

14V 1.0Hz Voltage Input
LPDT displacement (mm)
Current (A)
time (sec)

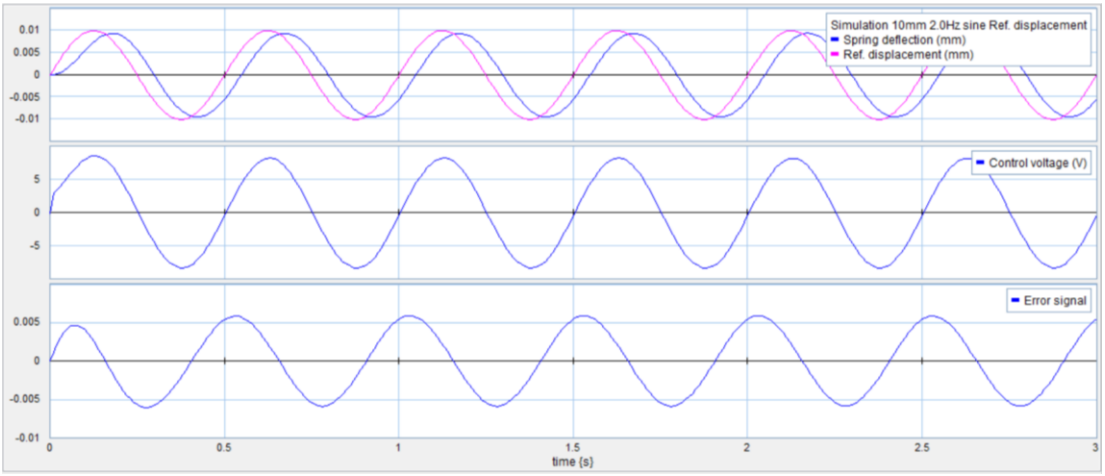

14V 1.0Hz Voltage Input; L=14mm
LPDT displacement (mm)
Current (A)
time (sec)

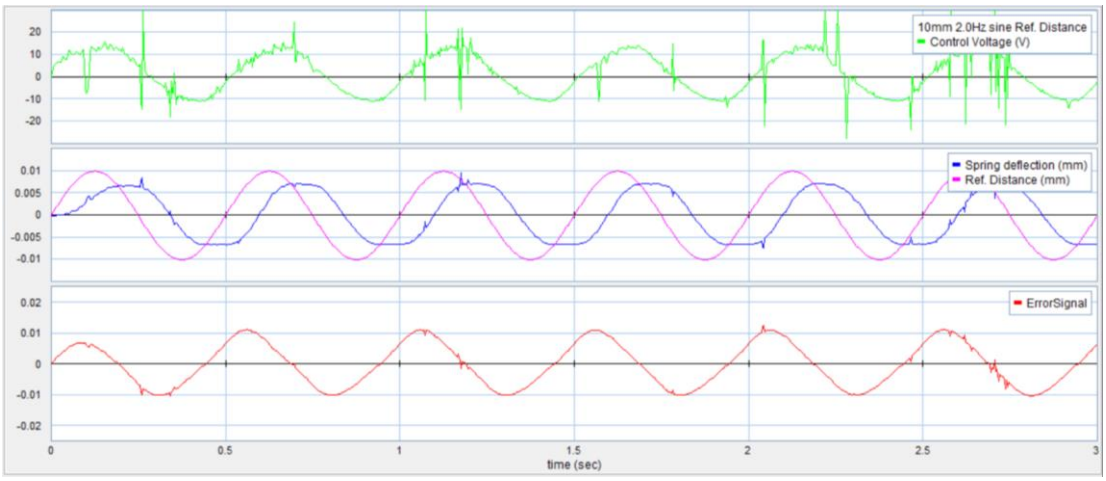

14V 1.0Hz Voltage Input; L=45mm
LPDT displacement (mm)
Current (A)
time (sec)

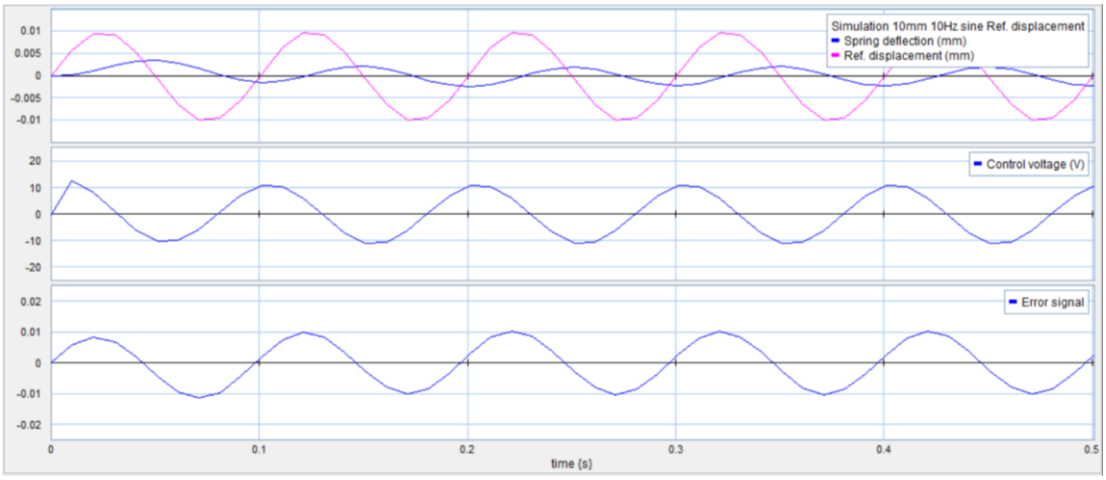
Closed Loop Position Control:

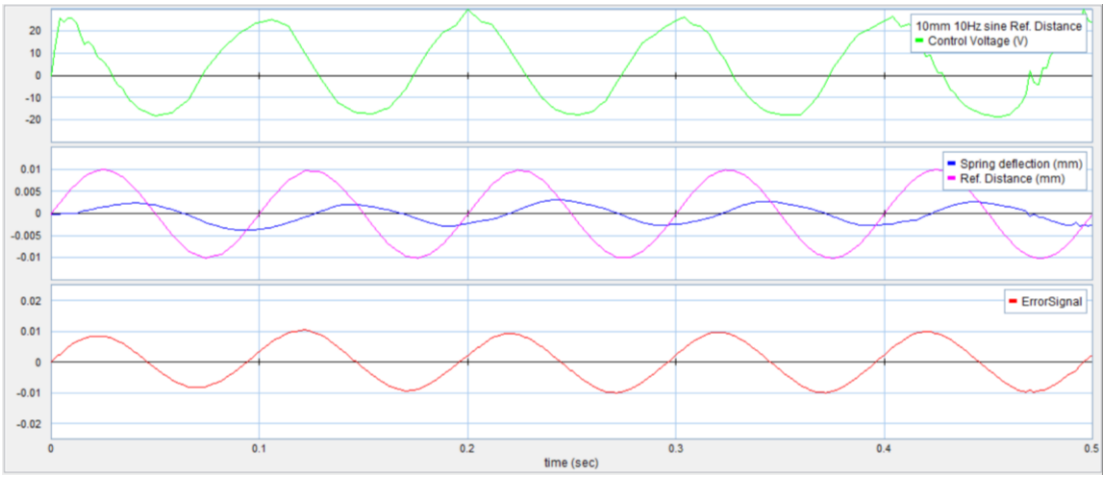




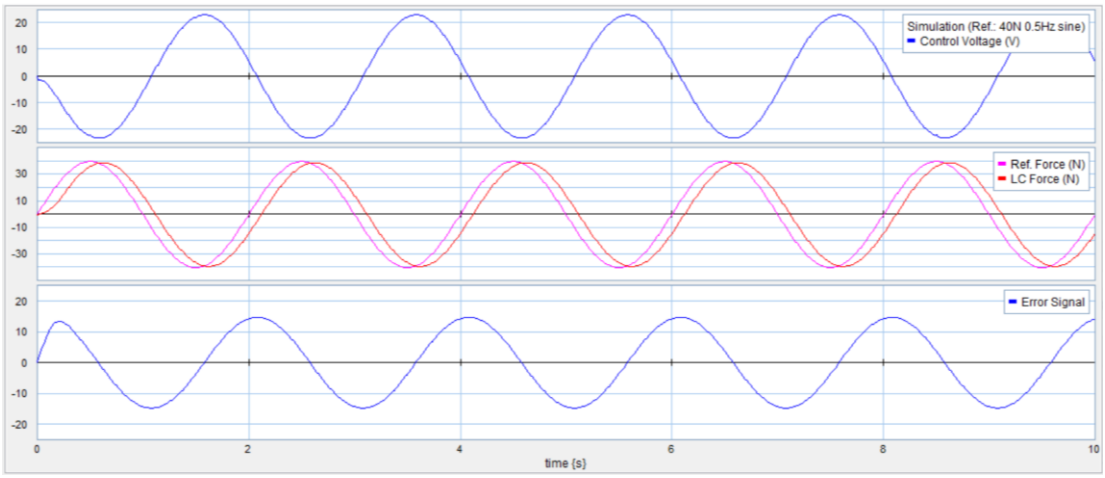


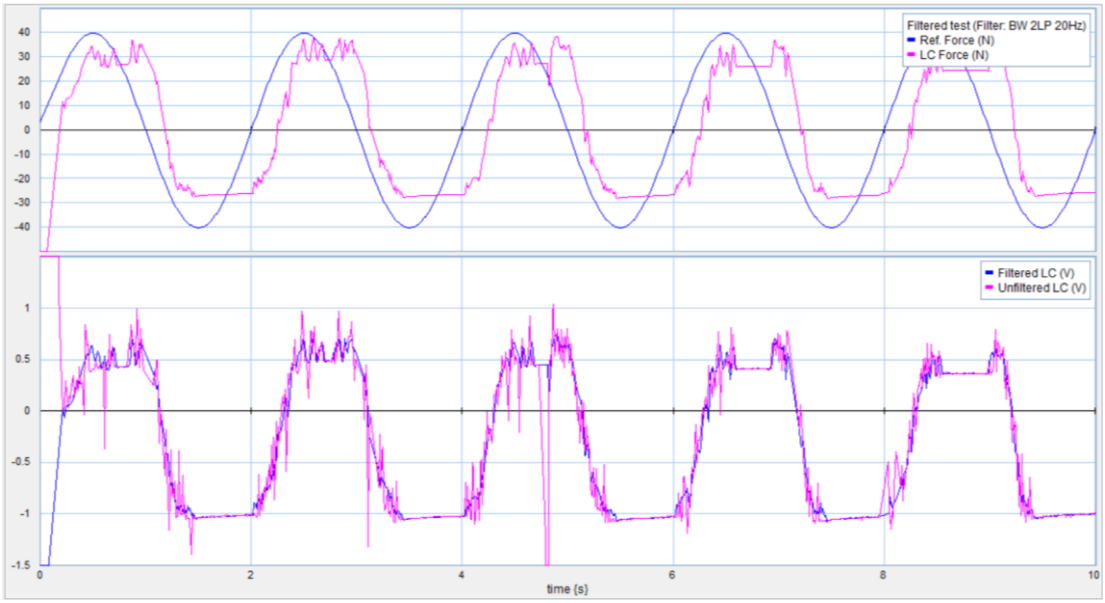

10mm 2.0Hz sine Ref. Distance
Control Voltage (V)
Spring deflection (mm)
Ref. Distance (mm)
ErrorSignal
time (sec)

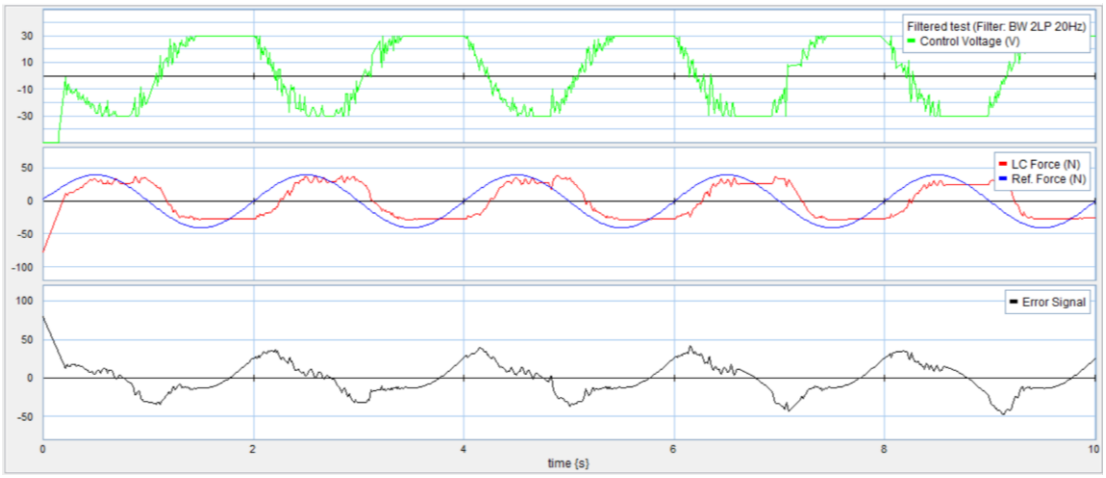

Simulation 10mm 10Hz sine Ref. displacement
Spring deflection (mm)
Ref. displacement (mm)
Control voltage (V)
Error signal
time (s)

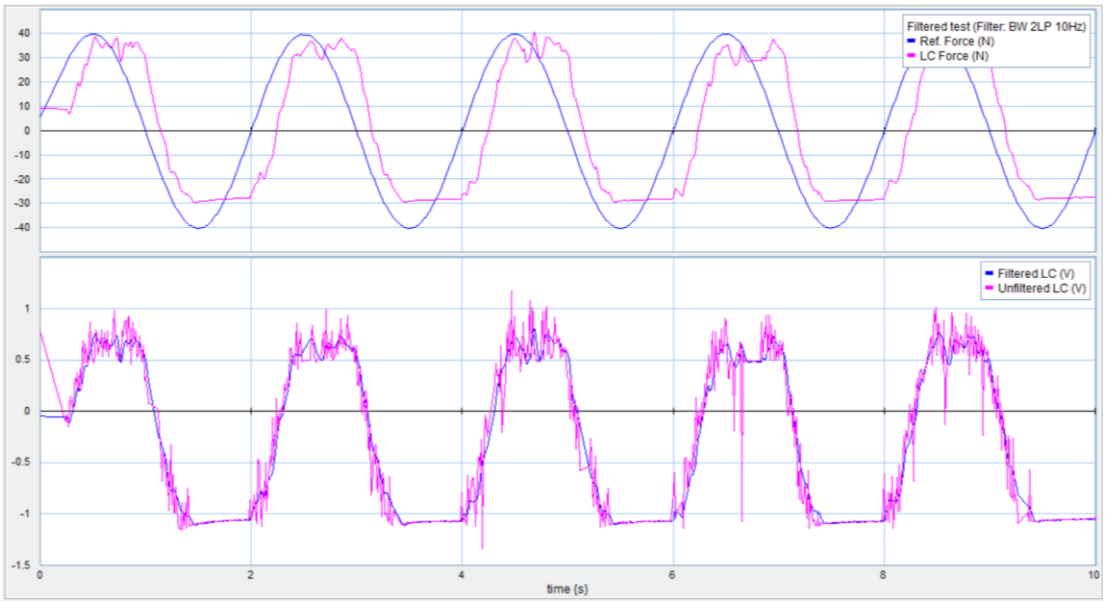

10mm 10Hz sine Ref. Distance
Control Voltage (V)
Spring deflection (mm)
Ref. Distance (mm)
ErrorSignal
time (sec)

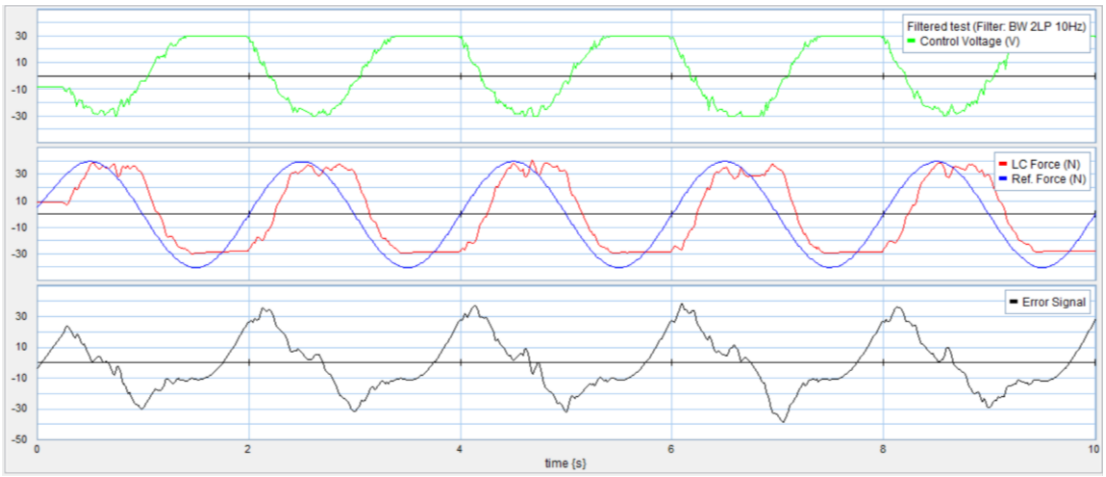
Closed Loop Force Control:

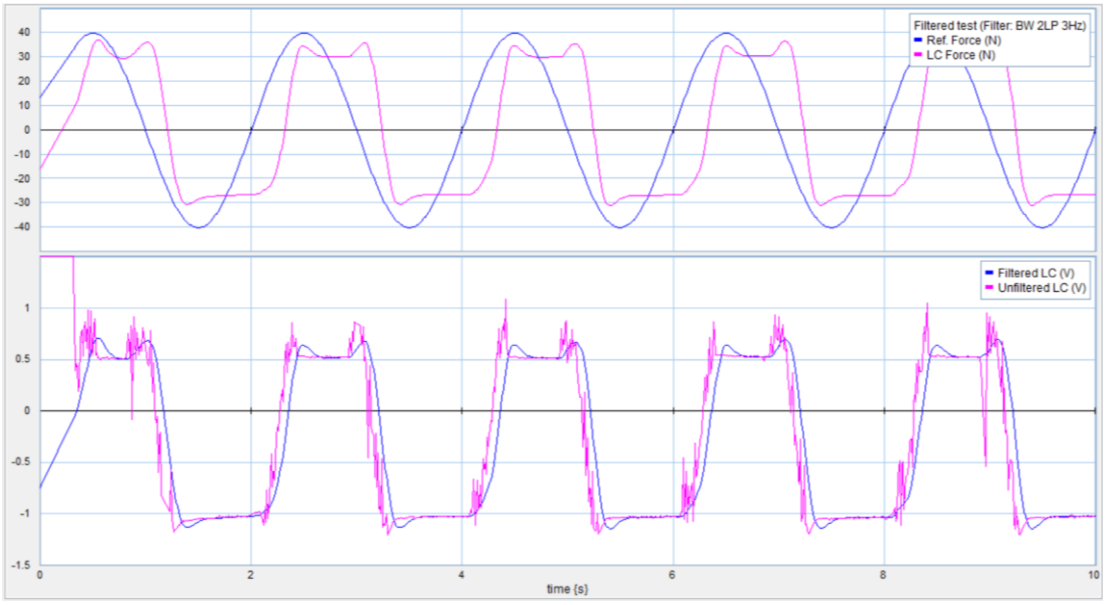


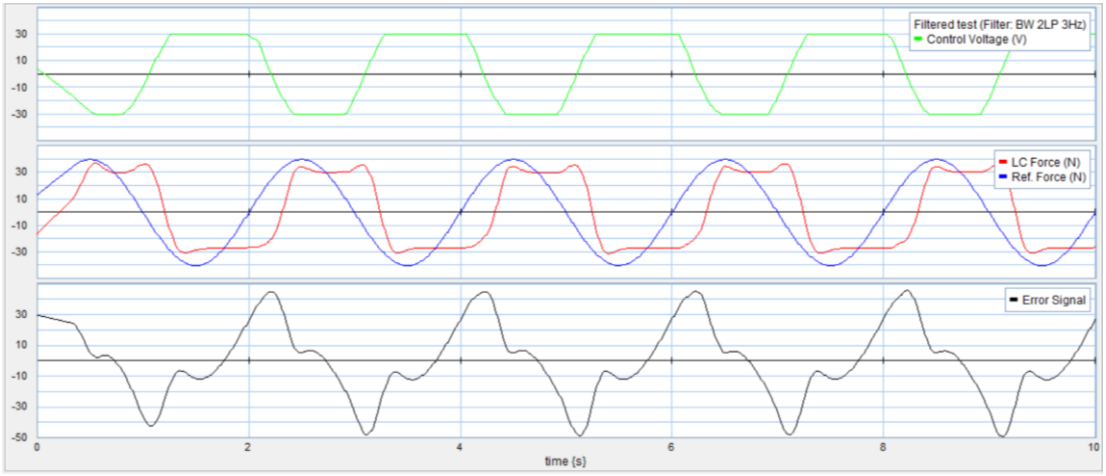



Filtered test (Filter: BW 2LP 20Hz)
Control Voltage (V)
LC Force (N)
Ref. Force (N)
Error Signal
time (s)


Filtered test (Filter: BW 2LP 10Hz)
Ref. Force (N)
LC Force (N)
Filtered LC (V)
Unfiltered LC (V)
time (s)


Filtered test (Filter: BW 2LP 10Hz)
Control Voltage (V)
LC Force (N)
Ref. Force (N)
Error Signal
time (s)


Filtered test (Filter: BW 2LP 3Hz)
Ref. Force (N)
LC Force (N)
Filtered LC (V)
Unfiltered LC (V)
time (s)


Filtered test (Filter: BW 2LP 3Hz)
Control Voltage (V)
LC Force (N)
Ref. Force (N)
Error Signal
time (s)

Road Profile (Pothole):